FDONT: FAST DETECTION OF NON-CROSSING TANDEMS

Done STOJANOV*, Aleksandra MILEVA

Faculty of Computer Science, University Goce Delcev, Krste Misirkov nn – Štip, Republic of Macedonia

**Corresponding author e-mail: done.stojanov@ugd.edu.mk*

Received 31 August 2017; accepted 4 December 2017

ABSTRACT

A new solution of the problem of computational tandem repeats detection is proposed. Non-crossing tandem repeats in genomic data are identified in few seconds on regular computer, due to computational jumps, out of tandems being already identified. Maximal left and right template extension is also employed. For practical use a desktop application was developed, allowing user to specify subject sequence for analysis and repeats' search parameters.

KEY WORDS: *fast, non-crossing, tandems, extension, jumps.*

INTRODUCTION

Tandem repeats are repetitions of short DNA patterns. While frequently found in non-coding regions, tandem repeats are rarely found in coding DNA. The polymorphic nature of these sequences makes them suitable for DNA profiling. Namely, it is unlikely that unrelated individuals will have similar VNTR (Variable Number Tandem Repeats) loci. On the other hand, VNTR loci among related individuals are highly similar, what makes them suitable for fingerprint analysis.

The process of computational analysis of repeated sequences in genomic material is not a trivial task due to the following reasons: the structure of the repeating pattern is un-known (very opposite to query-searching methodologies) and each copy may differs the previous in terms of mutations. Not only base substitutions, but also insertions (deletions) may occur.

Current methods for computational analysis of tandem repeats mutually differ in many ways.

First, there are methods that do not consider mutations and those that do consider mutations within repeats. Main's method (Main *et al.,* 1984) is suitable for identification of exact repeats (mutations are not considered) and had to have little application in practice, but its computational performance is superior. Instead of considering only exact repeats, Kolpakov (Kolpakov *et al.,* 2003) allows mismatches per copies. Methods such as: (Landau *et al.,* 2001), (Sokol *et al.,* 2001) that consider all types of mutations have more practical application in biology, but their computational performances are lower due to the combinatorial approaches employed.

Second, there are methods that can be applied only for detection of tandems with restriction to the length of the repeating unit. Rivals (Rivals *et al.,* 1997) proposed methodology that can be applied only to micro-satellites (less than 4 bp), while (Sagot *et al.,* 1998) can be applied to all types of tandem repeats. As expected, the more limited the application is, better computational performances are expected.

Third, some of the methods are deterministic, while other heuristic. The most popular application for this purpose: Tandem Repeat Finder (Benson, 1999) detects tandem of k-mers (copies of k nucleotides), but the search and the output depends of the user-specified parameters. On the other hand, STAR (Delgrange *et al.,* 2004) operates user-independently. This makes STAR less sensitive compared with Tandem Repeat Finder.

It is also important to notice that some methods such as (Stoye *et al.,* 2002) employ suffix tree data structure utilities in order to process tandem repeats. Suffix tree prime advances over other data structures is primarily in computational aspects since they ban be built/searched in linear time and they also require linear memory storage.

An extremely fast methodology for exact and approximate tandem repeats, considering also mismatches per copy, is proposed. It allows fast detection of non-crossing repeats, based on maximal extension approach and computational jumps once a tandem has been detected. This methodology was programmed as desktop application in C# environment and tested on complete *E. coli* genome. Obtained results showed that only a few seconds were required in order to detect all non-crossing micro-satellites in *E. coli* sample.

MATERIALS AND METHODS

Problem definition: *Given a DNA sequence* $X = \{x_i\}_{i=1}^{n}$ *over the alphabet* $\Sigma = \{A, C, T, G\}$ *we search for maximal repeats of type* $R = \{w_j\}_{j=1}^{T}, T \geq 2$ *in* X *such as each pair of words* (w_j, w_{j+1}) *is a pair of consecutive and non-overlapping words of* k *nucleotides in* X *and there is a word* w_ξ *in* R *such as the number of mismatching elements between* w_ξ *and* $w_z, z \neq \xi, 1 \leq z \leq T$ *in* R *is at most* m*. The repeat* R *is maximal if it can't be further extended to the left, right or both (to the left and right), i.e. there is no other longer repeat* R_l *in* X *that satisfies previous constrains, such as* R *is a substring of* R_l*.*

Let's briefly discuss introduced parameters and terms in the problem definition section. X is the sequence where we search for tandem repeats, k is the length of repeated template and m is the maximum number of mismatches allowed per repetition regarding the template. Since the structure of the template is not specified by the user, but only parameters $k$and$m$, each word of k nucleotides is a candidate for template that may repeat. A repeat is reported only if there is at least one adjacent copy of a template that differs at most m elements.

The difference between a repeat and maximal repeat we will discuss on the sample: $X = \mathrm{CTACGACGACAACCC}$ for $k = 3$and$m = 1$. The repeat $R\colon \mathrm{CTACGACGACAA}$ in X is maximal. This repeat can be spilt into four words of $k = 3$ nucleotides: $w_1\colon \mathrm{CTA}, w_2\,\mathrm{CGA}, w_3\colon \mathrm{CGA}, w_4\colon \mathrm{CAA}$. Words w_1, w_3 and w_4 differ 0 or 1 elements compared to w_2 if w_2 is chosen for template. The repeat R is maximal since it can't be further extended, because there are no elements prior w_1 in X and the triplet next to $w_4\colon \mathrm{CCC}$ differs w_2 more than $m = 1$ elements. Substrings such as: $w_2 w_3 w_4\colon \mathrm{CGACGACAA}, w_1\, w_2 w_3\colon \mathrm{CTACGACGA}, \ldots\ \mathrm{ect}$ are also repeats, but they are not maximal since further extension to the left, right, or the both is also possible in a way that the maximal repeat R can be obtained.

Our algorithm detects non-crossing tandem repeats, by taking words of k elements (k-mers) which are preceded and succeeded by at least k elements in X as candidates for template. The template is extended into repeat (longer string) if adjacent left (right) non-crossing word differ at most m elements regarding the candidate for template. While there are repeat-adjacent left (right) non-crossing words that differ at most m elements regarding the candidate for template, the repeat is further extended into larger repeat. It's important to note that we perform left-maximal extension first, then right-maximal extension. Once the maximal extension has been reached, i.e. a tandem repeat has been identified; the same procedure is repeated for k-mers out of the identified tandem repeat, also preceded and succeeded by at least k elements in$X$.

On the other hand, if neither left nor right extension is possible, there is no repeat. In such case the word that has $k-1$ crossing elements regards the current candidate for template is taken as a next candidate for template and the procedure is repeated.

This approach we will discuss on the sample: $X = \mathrm{AACAAAAAAAAACTCCCCCAA}$ for $k = 3$ and $m = 1$ first, then for$m = 0$. The first candidate for template is the leftmost word of $k = 3$ elements preceded by at least $k = 3$ nucleotides and succeeded by at least the same number of elements. In this case that is$X_{4-6} = \mathrm{AAA}$ which is preceded by AAC and succeeded by more than $k = 3$ nucleotides Fig. 1. The candidate for template can be left-extended in repeat, since it differ $X_{1-3} = \mathrm{AAC}$ one element. After this extension a repeat X_{1-6}: AACAAA is obtained, Fig. 1. Since further left-extension is not possible, we try right-maximal extension. Repeat right adjacent k-mer $X_{7-9} = \mathrm{AAA}$ hits perfectly the candidate for template$X_{4-6} = \mathrm{AAA}$ and further extended repeat X_{1-9}: $\mathrm{AACAAAAAA}$ is obtained. This is also true for $X_{10-12} = \mathrm{AAA}$that result in tandem repeatX_{1-12}: $\mathrm{AACAAAAAAAAA}$ Fig. 1 This is a maximal repeat, because the next k-mer $X_{13-15} = \mathrm{CTC}$ differs the template more than $m = 1$ elements.

According to the employed methodology the next candidate for template is taken out of the repeat and it should be preceded and followed by at least 3 elements. That is$X_{16-18} = \mathrm{CCC}$ Fig. 1. If the same procedure is applied in the remaining space out of the previously identified tandem repeatX_{1-12}, that would result with detection of second tandem repeat$X_{13-18} = \mathrm{CTCCCC}$. If $m = 0$ only one ETR (exact tandem repeat):$X_{4-12} = \mathrm{AAAAAAAAA}$ is reported.

Now lets' substitute the fifth and the sixth element in X with G (Guanine) $X = \mathrm{AACAGGAAAAAACTCCCCCAA}$ and consider the problem once again for $k = 3$ and$m = 1$. The first candidate for template $X_{4-6} = \mathrm{AGG}$ can be neither left nor right extended, because $X_{1-3} = \mathrm{AAC}$ and $X_{7-9} = \mathrm{AAA}$ mismatch the candidate more than $m = 1$ nucleotide Fig. 2. In such case, the next candidate for template is the word that has

$k-1=2$ elements in common (GG) with the previous candidate for template and in this case that is$X_{5-7} = \mathrm{GGA}$ Fig. 2. Neither left, nor right extension is also possible for $X_{5-7} = \mathrm{GGA}$ and that's why $X_{6-8} = \mathrm{GAA}$ is taken for next candidate for template Fig. 2. For this candidate right-extension by $X_{9-11} = \mathrm{AAA}$ is possible, that results into repeat$X_{6-11} = \mathrm{GAAAAA}$, Fig. 2.

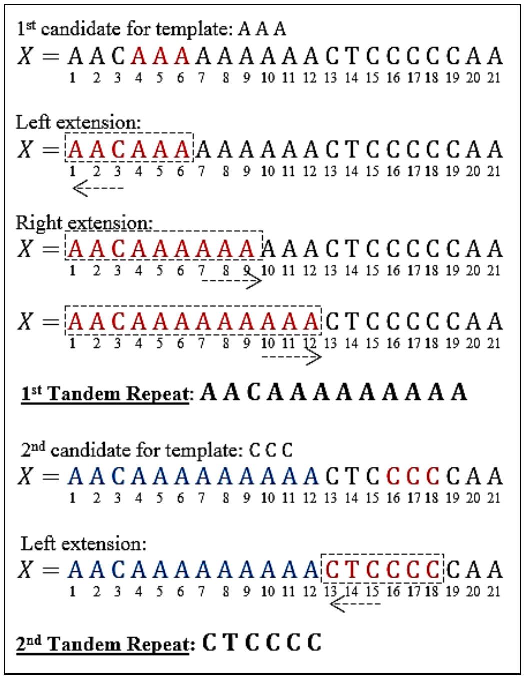

1st candidate for template: A A A
X = A A C A A A A A A A A A C T C C C C C A A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
Left extension:
X = A A C A A A A A A A A A C T C C C C C A A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
Right extension:
X = A A C A A A A A A A A A C T C C C C C A A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
X = A A C A A A A A A A A A C T C C C C C A A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
1st Tandem Repeat: A A C A A A A A A A A A
2nd candidate for template: C C C
X = A A C A A A A A A A A A C T C C C C C A A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
Left extension:
X = A A C A A A A A A A A A C T C C C C C A A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
2nd Tandem Repeat: C T C C C C

FIG. 1. Searching for tandem repeats, first candidate first repeat

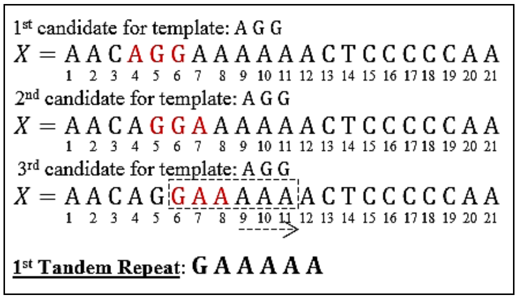
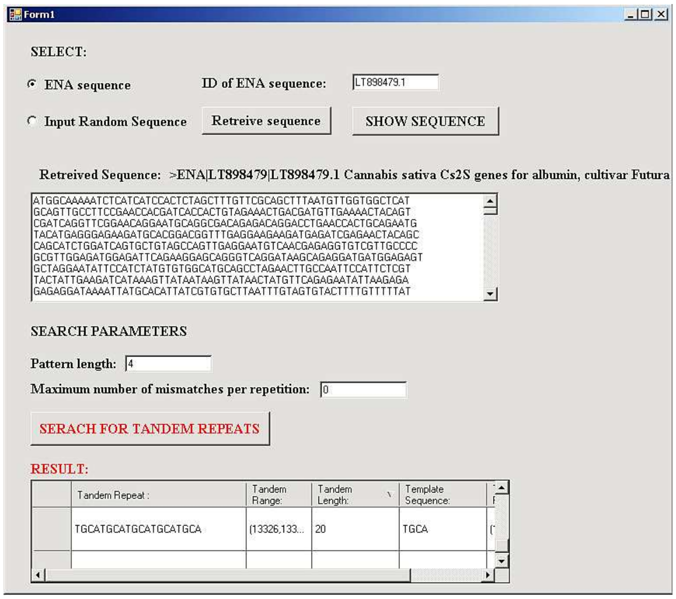

1st candidate for template: A G G
X = A A C A G G A A A A A A C T C C C C C A A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
2nd candidate for template: A G G
X = A A C A G G A A A A A A C T C C C C C A A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
3rd candidate for template: A G G
X = A A C A G G A A A A A A C T C C C C C A A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
1st Tandem Repeat: G A A A A A

FIG. 2. Searching for tandem repeats, first candidate no repeat

Due to the jumps out of the already identified repeats this methodology allows ultra-fast detection of tandem repeats. By taking each crossing word as a possible candidate for template, crossing tandem repeats could be also detected, but that will significantly worsen the computational performance, limiting the application only to small genomic sequences.

An implementation of the proposed methodology as a desktop application in Microsoft Visual C # 2008 Express Edition was programmed Fig. 3. This application accepts as input: random sequence for analysis or retrieves samples from ENA (European Nucleotides Archive) based on the ID provided by the user. Existing and successfully retrieved sequences from ENA in FASTA data format are shown in the upper textbox control, Fig. 3. Once the sequence has been retrieved or random sample was provided by the user, the search parameters: k and m have also to be specified. Pattern length (k) is the length of the repeating template we search for and the maximum number of mismatches per repetition allowed is in fact the parameter m, Fig. 3. The results of the search are shown in dataGridView control, including: tandem's structure, tandem's range, tandem's length and details regarding the template sequence.



SELECT:

● ENA sequence ID of ENA sequence: LT898479.1

○ Input Random Sequence

Retreived Sequence: >ENA|LT898479|LT898479.1 Cannabis sativa Cs2S genes for albumin, cultivar Futura

```
ATGGCAAAAATCTCATCATCCACTCTAGCTTTGTTCGCAGCTTTAATGTTGGTGGCTCAT
GCAGTTGCCTTCCGAACCACGATCACCACTGTAGAAACTGACGATGTTGAAAACTACAGT
CGATCAGGTTCGGAACAGGAATGCAGGCGACAGAGACAGGACCTGAACCACTGCAGAATG
TACATGAGGGAGAAGATGCACGGACGGTTTGAGGAAGAAGATGAGATCGAGAACTACAGC
CAGCATCTGGATCAGTGCTGTAGCCAGTTGAGGAATGTCAACGAGAGGTGTCGTTGCCCC
GCGTTGGAGATGGAGATTCAGAAGGAGCAGGGTCAGGATAAGCAGAGGATGATGGAGAGT
GCTAGGAATATTCCATCTATGTGTGGCATGCAGCCTAGAACTTGCCAATTCCATTCTCGT
TACTATTGAAGATCATAAAGTTATAATAAGTTATAACTATGTTCAGAGAATATTAAGAGA
GAGAGGATAAAATTATGCACATTATCGTGTGCTTAATTTGTAGTGTACTTTTGTTTTTAT
```

SEARCH PARAMETERS

Pattern length: 4

Maximum number of mismatches per repetition: 0

RESULT:

	Tandem Repeat :	Tandem Range:	Tandem Length:	Template Sequence:	F
	TGCATGCATGCATGCATGCA	(13326,133...	20	TGCA	(

FIG. 3. User interface

RESULTS AND DISCUSSIONS

Tests on *Escherichia coli strain AR_0149*, complete genome (ENA id: *CP021532.1*) of approximately 5 Mb (mega bases) were performed on Acer Aspire 5570Z computer with Genuine Intel CPU at 1,73 GHz and 2 GB RAM.

Screening the genome for non-crossing simple sequence repeats (SSRs) of k: 3, 4 and 5 (bp), with maximum m: 0, 1 and 2 mismatches allowed per repetition, results shown in

Table 1 were obtained. For each combination (k, m) the total number of tandem repeats, tandems' size and the running time of the application were recorded Table 1.

Let's first analyze the number of tandem repeats regards the parameters k and m. Shorter SSRs were screened, the more non-crossing repeats were detected Table 1. Screening for ETRs (exact tandem repeats) of 3, 4 and 5 bp, totally: 72856, 12595 and 3233 were found respectively Table 1, Fig. 4. On the other hand, as expected, the number of recorded tandems drastically increased while screening for ATRs (approximate tandem repeats), if maximum $m: 1{,}2$ mismatches were allowed per SSR repetition. In general, by increasing the parameter m, the number of ATRs also increased.

In terms of the tandems' size, the longest approximate tandems repeats were found for the shortest templates. Screening for 3bp SSRs with maximum 2 mismatches allowed per repetitions, tandems of size ranging between: 6 and 102 bps of period 3 (6, 9, 12, 15, 96, 99, 102) were found. Variation in exact repeats' size is higher for shorter SSRs, Fig. 5.

Screening the sample for exact tandems of 5 bp required less time than screening for exact tandems of 4 bp and this required less time than screening the sample for exact tandems of 3 bp, Fig. 6. As expected, the time requirement for detection of approximate tandems is higher than the time required to find exact repeats, since more tandems are processed.

TABLE 1. Results of searching *E. coli* strain *AR_0149* for ETR and ATR for different combinations of (k,m)

Sequence	k	m	Number of Tandem Repeats	Tandems' Length	Time (s)
Escherichia coli strain AR_0149, complete genome	3	0	72856	6,9,12,15	18,653
		1	307754	6,9,12,15,18,21,24,27,30,33	70,145
		2	396387	6 – 102 (+3)	92,208
(4,830,167 bp)	4	0	12595	8,12	5,286
		1	104287	8,12,16,20,24,28	26,382
		2	285730	8 – 56 (+4)	50,511
(ID: CP021532.1)	5	0	3233	10	3,554
		1	33882	10,15,20,25	10,220
		2	136285	10,15,20,25,30,35,40	27,680

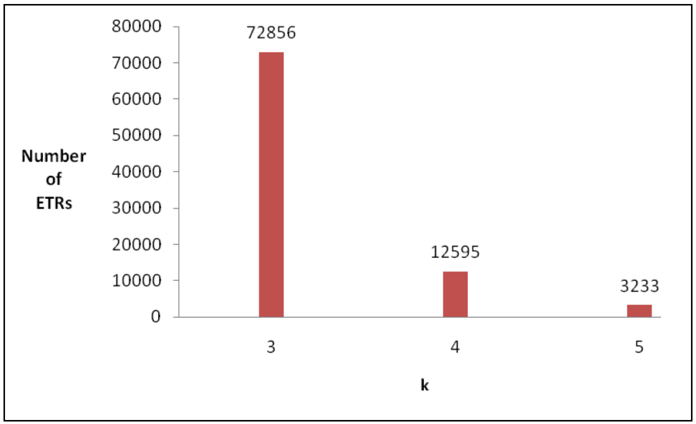


FIG. 4. Number of exact tandem repeats for $k = 3,4,5$

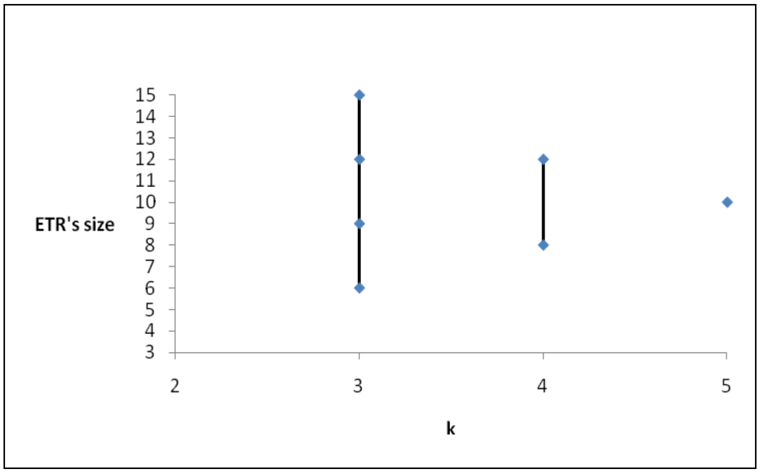


FIG. 5. Variability in exact tandems' size for $k = 3, 4, 5$

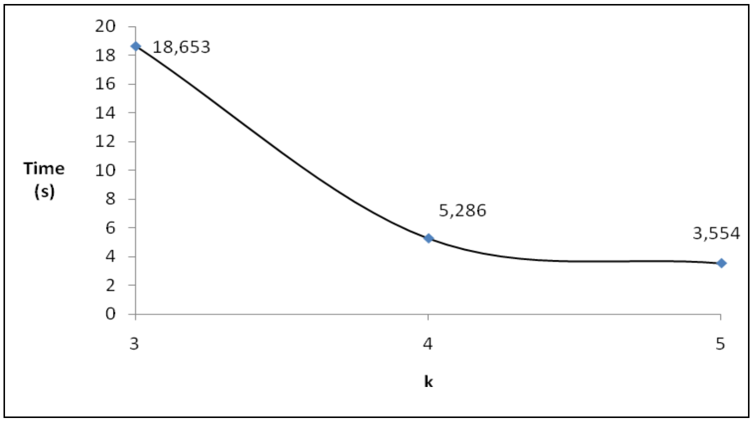


FIG. 6. Running time for exact tandems detection for $k = 3, 4, 5$

CONCLUSIONS

A new methodology for analysis of tandem repeats was proposed. Applying this methodology non-crossing exact and approximate tandem repeats can be detected. For practical purpose, a desktop application in C# was developed. This application allows repeats' detection and analysis of random user-provided samples or genomic sequences retrieved from ENA (European Nucleotide Archive) according to user-provided search parameters. Due to the employed computational strategy, this application is applicable on whole genomes on standard computer, generating results in just a few seconds.

REFERENCES

- Benson G.1999.Tandem repeats finder: a program to analyze DNA sequences. *Nucleic acids research. 27*(2):573-580.
- Benson G. 2005.Tandem cyclic alignment. *Discrete applied mathematics*. 146(2):124-133.
- Delgrange O., Rivals E.2004.STAR: an algorithm to search for tandem approximate repeats. *Bioinformatics.* 20(16):2812-2820.
- Fischetti VA., Landau GM., Sellers PH., Schmidt JP.1993.Identifying periodic occurences of a template with applications to protein structure. *Information Processing Letters*. 45(1):11-18.
- Groult R., Léonard M., Mouchard L. 2004. Speeding up the detection of evolutive tandem repeats. *Theoretical computer science.* 310:309-328.
- Hammock EA., Young LJ.2005. Microsatellite instability generates diversity in brain and sociobehavioral traits. *Science.* 308(5728):1630-1634.
- Kolpakov R., Kucherov G.1999. Finding maximal repetitions in a word in linear time, pp. 596-604. In: Foundations of Computer Science, 40th Annual Symposium on. IEEE.
- Kolpakov R., Kucherov G. 2003. Finding approximate repetitions under Hamming distance. *Theoretical Computer Science.* 303(1):135-156.
- Landau GM., Schmidt JP., Sokol D. 2001. An algorithm for approximate tandem repeats. *Journal of Computational Biology.* 8(1):1-18.
- Main MG., Lorentz RJ.1984. An O(nlogn) algorithm for finding all repetitions in a string. *Journal of Algorithms.* 5(3):422-432.
- O'Dushlaine C., Shields DC. 2006.Tools for the identification of variable and potentially variable tandem repeats. *BMC genomics.* 7(1):290.
- Rivals E., Delgrange O., Delahaye JP., Dauchet M., Delorme MO., Hénaut A., Ollivier E. 1997. Detection of significant patterns by compression algorithms: the case of approximate tandem repeats in DNA sequences. *Bioinformatics.* 13(2):131-136.
- Reneker J., Shyu CR., Zeng P., Polacco JC., Gassmann W. 2004. ACMES: fast multiple-genome searches for short repeat sequences with concurrent cross-species information retrieval. *Nucleic acids research.* 32(suppl_2):W649-W653.
- Sagot MF., Myers EW. 1998. Identifying satellites and periodic repetitions in biological sequences. *Journal of Computational Biology.* 5(3):539-553.
- Stoye J., Gusfield D. 2002. Simple and flexible detection of contiguous repeats using a suffix tree. *Theoretical Computer Science.* 270(1-2):843-56.
- Sobreira TJ., Durham AM., Gruber A. 2005.TRAP: automated classification, quantification and annotation of tandemly repeated sequences. *Bioinformatics.* 22(3):361-362.
- Sokol D., Benson G., Tojeira J. 2007.Tandem repeats over the edit distance. *Bioinformatics.* 23(2):e30-e35.
- Taneda A. 2004. Adplot: detection and visualization of repetitive patterns in complete genomes. *Bioinformatics.* 20(5):701-708.