A Bioinformatics Approach for Detecting Repetitive Nested Motifs using Pattern Matching

Libertas Academica
FREEDOM TO RESEARCH

José R. Romero[1,†], Jessica A. Carballido[2], Ingrid Garbus[1,3], Viviana C. Echenique[1,4] and Ignacio Ponzoni[2]

[1]Centro de Recursos Naturales Renovables de la Zona Semiárida (CERZOS) – CONICET, Bahía Blanca, Argentina. [2]Instituto de Ciencias e Ingeniería de la Computación (ICIC) – CONICET, Universidad Nacional del Sur, Bahía Blanca, Argentina. [3]Departamento de Ciencias de la Salud, Universidad Nacional del Sur, Bahía Blanca, Argentina. [4]Departamento de Agronomía, Universidad Nacional del Sur, Bahía Blanca, Argentina. [†]Deceased on 26 February 2016.

ABSTRACT: The identification of nested motifs in genomic sequences is a complex computational problem. The detection of these patterns is important to allow the discovery of transposable element (TE) insertions, incomplete reverse transcripts, deletions, and/or mutations. In this study, a de novo strategy for detecting patterns that represent nested motifs was designed based on exhaustive searches for pairs of motifs and combinatorial pattern analysis. These patterns can be grouped into three categories, motifs within other motifs, motifs flanked by other motifs, and motifs of large size. The methodology used in this study, applied to genomic sequences from the plant species *Aegilops tauschii* and *Oryza sativa*, revealed that it is possible to identify putative nested TEs by detecting these three types of patterns. The results were validated through BLAST alignments, which revealed the efficacy and usefulness of the new method, which is called Mamushka.

KEYWORDS: repetitive motifs, nested motifs, exact sequence analysis, structural bioinformatics

CITATION: Romero et al. A Bioinformatics Approach for Detecting Repetitive Nested Motifs using Pattern Matching. *Evolutionary Bioinformatics* 2016:12 247–251 doi: 10.4137/EBO.S40138.

TYPE: Original Research

RECEIVED: May 23, 2016. **RESUBMITTED:** September 12, 2016. **ACCEPTED FOR PUBLICATION:** September 15, 2016.

ACADEMIC EDITOR: Liuyang Wang, Associate Editor

PEER REVIEW: Four peer reviewers contributed to the peer review report. Reviewers' reports totaled 1397 words, excluding any confidential comments to the academic editor.

FUNDING: This work has been funded by grants PIP112-200801-01517, PIP112-2009-0100322, International Coop. Resolution D3077/12, ANPCyT PICT 2012-0660, PGI 24/A168, PGI 24/N032, PGI 24/ZN26, PGI 24/ZN21 and 24TA/11. The authors confirm that the funder had no influence over the study design, content of the article, or selection of this journal.

COMPETING INTERESTS: No statement regarding his competing interests, if any, was received from José R. Romero. Other authors disclose no potential conflicts of interest.

CORRESPONDENCE: ip@cs.uns.edu.ar

Paper subject to independent expert blind peer review. All editorial decisions made by independent academic editor. Upon submission manuscript was subject to anti-plagiarism scanning. Prior to publication all authors have given signed confirmation of agreement to article publication and compliance with all applicable ethical and legal requirements, including the accuracy of author and contributor information, disclosure of competing interests and funding sources, compliance with ethical requirements relating to human and animal study participants, and compliance with any copyright requirements of third parties. This journal is a member of the Committee on Publication Ethics (COPE).

Published by Libertas Academica. Learn more about this journal.

Introduction

Repetitive sequences contribute to genome structure, function, and evolution. They are classified into transposable elements (TEs), tandem repeats, and high copy number genes.[1] TEs, defined as DNA fragments with structural and functional characteristics that allow movement throughout the genome, are the most abundant component of many genomes.[2] For example, almost 80% of the *Triticum aestivum* genome is composed of TEs,[3] while TEs represent approximately 66% of the *Aegilops tauschii* genome[4] and about 35% of the *Oryza sativa* genome.[5] TE transpositions often produce mutations or changes in genome size[6]; thus, TEs can have a large impact on the evolution of a species.[7] One of the most abundant TEs in the plant kingdom are the long terminal repeat (LTR) retrotransposons, mainly consisting of a polyprotein coding sequence flanked by LTRs at the 5′ and 3′ ends.[1–3]

The existing tools for the detection of TEs rely on the following four types of approaches: de novo, structure-based, homology-based, and comparative genomics (reviewed in the study by Bergman and Quesneville[6]). In the de novo approach, the most common strategy detects all pairs of similar sequences at different locations within the analyzed input sequence, by comparing the input sequence to itself, concomitantly allowing the discovery and classification of repetitions such as TEs, tandem repeats, segmental duplication, and satellites. The use of this search method requires high-quality assemblies.[8] The second approach for TE identification uses prior knowledge of the common structures shared by different TEs. This method searches only for structures that have been previously reported. As reference databases do not include nested TEs, it is almost impossible to identify them using this approach.[9] The third approach involves searching for homology, usually based on a heuristic comparison within a sequence database.[10] In this case, if a TE is located inside another TE, the method reports only the internal element. Finally, the comparative genomics approach describes a group of methods that rely on neither homology nor structural features but detect new families of TEs based on the fact that TE transposition causes large insertions that can be detected with multiple sequence alignment. These methods search for insertion regions where multiple alignments of orthologous genome sequences are disrupted by a large insertion in one or more species.[11] Again, under this approach, a nested element will be reported only if it emerges from the comparison with some other genome.[6]

Overall, these methods allow for relatively easy detection of TEs that lack mutations, insertions, or deletions that alter their sequences. However, in nature, individual TEs have great inherent biological complexity, with temporal divergence between sequences and the possible insertion of TEs inside other TEs, thus interrupting their structure. In this study, we propose a method to identify nested motifs (ie, a subsequence of the input sequence that is repeated two or more times and might correspond to TEs nested into other TEs) that may provide an advantage over conventional TE search methods that are not specifically designed to identify nested motifs.

The main objectives of this study were to identify perfect motifs repeated at any distance within the genome and detect the patterns of nested motifs. This approach is based on the fact that the LTRs located at each end of the TEs are almost perfect repeats.

Materials and Methods

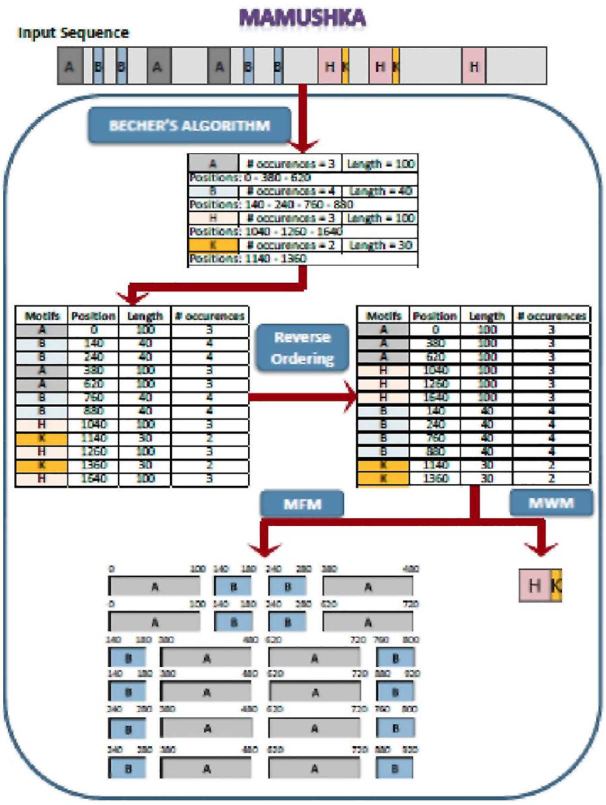
Developing a tool to identify individual TEs is intricate given the inherent biological complexity of this problem, the divergence between the sequences, the presence of incomplete reverse transcripts, and the existence of nested sequences within other TEs.[12] Moreover, it is difficult to define the boundaries of an element since TE sequences with long deletions or insertions are also present in genomes. To address this complexity, we propose a de novo method that starts searching for perfect motifs at any distance. Then, an exhaustive search is performed between the locations where both motifs appear, to identify cases of a TE inside another one. In this context, this tool will allow analysis of the structures of insertions of different families of TEs by means of looking for specific features that can define the behavior of a TE family. Therefore, in addition to the initial identification of a list of putative motifs, the detection of TEs is addressed by a combinatorial pattern analysis approach.

Proposed approach. Our proposed approach is called Mamushka since its functioning resembles the Russian wooden nesting doll, a set of dolls of decreasing size placed one inside another. Mamushka uses Becher's et al algorithm[13] as the first preprocessing step. Becher's et al method performs a search for all perfect repeats in the genome based on the suffix array construction by Manber and Myers.[14] This algorithm uses the nucleotide sequence as input data, up to 500 million nucleotide bases, and a lower bound for the length of the patterns to be reported. The method returns a list where each element contains two lines; in the first line, the pattern is described together with its number of occurrences and its length, and in the second line the positions of each occurrence are listed. There is no upper bound for the length of perfect repeats and the occurrences can be at any distance from each other.

In the next step, a list (L) of *t*-tuples (with $t = 4$) is built where each element of L is transformed from each element returned by Becher's et al algorithm. For each tuple, the first column stores a perfect repeat motif, the second column shows the position where the perfect repeat motif begins, the third column contains its length, and the fourth column shows the number of repetitions of the motif in the input file. Once this list of tuples is constructed, the elements are sorted based on decreasing length (column 3), thus building a new ordered list that is used to make the subsequent searches for both motifs within motifs and two identical motifs flanked by two other identical motifs (Fig. 1).

The final step of Mamushka is divided into two sub-algorithms called motifs within motifs (MWM) and motifs flanked by other motifs (MFM). For MWM, given two motifs, the objective is to determine whether one of the motifs is perfectly contained inside the other motif. P and Q stand for two motifs that belong to the L ordered list, *i* is defined as the starting position of P and *j* is the ending position of P obtained by adding P's length to *i*. *k* and *l* are analogously defined for motif Q (Fig. 2A). To reduce the size of the exhaustive search, pairs with Q greater in length than P are not considered. Q should appear at least as many times as P. Then, once $P_i \le Q_k \le Q_l \le P_j$ is verified, it can be confirmed that Q is a motif inside P (Fig. 2A).

The second sub-algorithm searches for motifs flanked by other motifs. M, N, and target site duplication (TSD)



A	# occurences = 3	Length = 100
Positions: 0 - 380 - 620		
B	# occurences = 4	Length = 40
Positions: 140 - 240 - 760 - 880		
H	# occurences = 3	Length = 100
Positions: 1040 - 1260 - 1640		
K	# occurences = 2	Length = 30
Positions: 1140 - 1360		

Motifs	Position	Length	# occurences
A	0	100	3
B	140	40	4
B	240	40	4
A	380	100	3
A	620	100	3
B	760	40	4
B	880	40	4
H	1040	100	3
K	1140	30	2
H	1260	100	3
K	1360	30	2
H	1640	100	3

Motifs	Position	Length	# occurences
A	0	100	3
A	380	100	3
A	620	100	3
H	1040	100	3
H	1260	100	3
H	1640	100	3
B	140	40	4
B	240	40	4
B	760	40	4
B	880	40	4
K	1140	30	2
K	1360	30	2

Figure 1. General layout of the Mamushka method. The pipeline of the proposed new method to detect perfect repeat sequences is illustrated. The identified motifs are illustrated in the input sequence scheme. The frame encloses the steps followed by the method leading to a list of motifs within motifs and motifs flanked by other motifs.

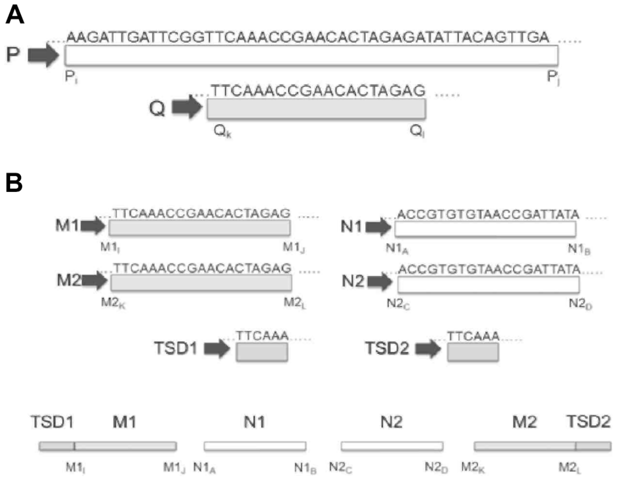

A
P
...AAGATTGATTCGGTTCAAACCGAACACTAGAGATATTACAGTTGA .....
P_i
P_j
Q
...TTCAAACCGAACACTAGAG .....
Q_k
Q_l
B
M1
...TTCAAACCGAACACTAGAG .....
$M1_I$
$M1_J$
N1
...ACCGTGTGTAACCGATTATA .....
$N1_A$
$N1_B$
M2
...TTCAAACCGAACACTAGAG .....
$M2_K$
$M2_L$
N2
...ACCGTGTGTAACCGATTATA.....
$N2_C$
$N2_D$
TSD1
...TTCAAA.....
TSD2
...TTCAAA.....
TSD1
M1
N1
N2
M2
TSD2
$M1_I$
$M1_J$
$N1_A$
$N1_B$
$N2_C$
$N2_D$
$M2_K$
$M2_L$

Figure 2. Mamushka rationale. (**A**) Motifs within other motifs. P and Q represent two motifs from the L ordered list, *i* and *k*: the starting position of P and Q, respectively; *j* and *l*: the ending position of P and Q, respectively. Q is a motif inside P if $P_i \leq Q_k \leq Q_l \leq P_j$. (**B**) Motifs flanked by other motifs starting with two motifs from the list L. M1, M2, N1, N2, TSD1, and TSD2 are three pairs of perfect distinct motifs. Then, the search determines whether N is flanked by other pairs of perfect motifs. I and K: the starting position of M1 and M2; J and L: the end position of M1 and M2; A and C: the starting position of N1 and N2; B and D: the end position of N1 and N2. If $M1_I \leq M1_J \leq N1_A \leq N1_B \leq N2_C \leq N2_D \leq M2_K \leq M2\ _L$ the algorithm continues the search for TSD elements that should end at the *i*–1 position, and TSD2 should start at the L+1 position.

denote three perfect distinct motifs taken from the L ordered list (Fig. 2B). The search determines whether N is flanked by other pairs of perfect motifs. M1 and M2 are defined to share motif M in different positions of the input sequence, as N1 and N2 share motif N and TSD1 and TSD2 share motif TSD. I and K denote the starting position of M1 and M2, and J and L denote their end; in addition, A and C denote the starting positions of N1 and N2, and B and D denote their final positions (Fig. 2B). Then, after verifying that $M1_I \leq M1_J \leq N1_A \leq N1_B \leq N2_C \leq N2_D \leq M2_K \leq M2_L$, the algorithm continues the search for TSD elements. TSD1 should end at the *i*–1 position and TSD2 should start at the L+1 position. These motifs, which define a TSD type, are also found in the L ordered list with perfect repetitions between 4 and 6 bp.

The use of big-O notation allows the classification of the algorithms based on how they respond, specifically describing the worst-case scenario. Both MWM and MFM are quadratic time algorithms $O(n^2)$, meaning that their performance is directly proportional to the square of the size of the input data set. In the case of these algorithms, the input is the list of repetitive motifs as shown in Figure 1. Algorithms presenting quadratic time are considered tractable algorithms, meaning that their running times are computationally reasonable.

Additional technical details and software availability. Our Mamushka tool was written in Awk[15] and is available at the following website: http://lidecc.cs.uns.edu.ar/mamushka. To provide easy-to-use software, the code runs through a terminal console. The front-end of this script was created by invoking it from a website. Execution functions called EXEC were used to execute an external program, the Awk script. Then, for statistical purposes, an R-script[16] was programmed to show the distribution of the motifs and the number of the repetitions. After showing the distribution plot, the program allows the user to choose the range for the motifs, and motifs outside of this range will be discarded. After confirming the range, the program will perform another invocation of the code, this time calculating for motifs within other motifs, motifs flanked by other motifs, and large motifs within the stipulated size. This calculation will return three separate files that can be downloaded individually in LinDna format.

Validation of the Proposed Mamushka Method for the Detection of Nested Repetitive Elements

Given the deterministic and exhaustive exploratory features of the method, testing with artificial data would be trivial; accordingly, only real biological sequences were selected for the validation of Mamushka algorithm. By doing this, it was possible to demonstrate that the method can identify bona fide nested repetitive elements, even when the strategy searches for perfect repeated motifs. Thus, the experiment was conducted using DNA sequences from *O. sativa* and *Ae. tauschii* obtained from NCBI databases, included in the website as application examples 1 and 2, respectively. The output list containing the distribution of all perfect motifs found, grouped according to the number of times each motif was repeated, should help the user to select a search threshold. Then, Mamushka begins searching for motifs within motifs, motifs flanked by other motifs, and remote long motifs that comply with the chosen threshold.

Application example 1. The entire *O. sativa* chromosome 10 (23.7 Mb)[5] was used as input data. The distribution of perfect repeat motifs revealed that there was a sufficient number of motifs with lengths ranging from 10 to 17 bp (Fig. 3), showing a peak that continuously decreased. After the fulfillment

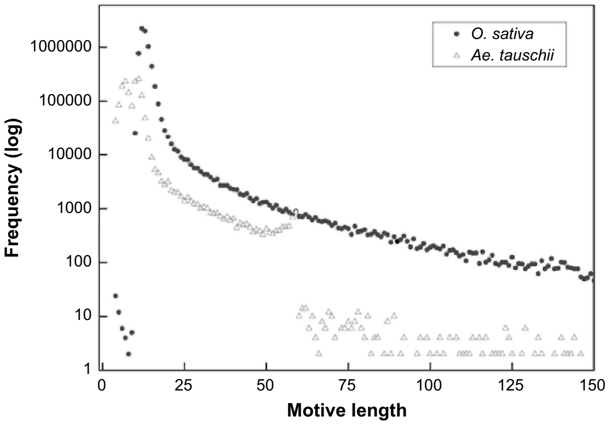


Figure 3. Frequency of various motif sizes. The frequency of the motifs was classified according to the size for (**A**) *O. sativa* chromosome 10 (23.7 Mb) (*l*), and (**B**) *Aegilops tauschii* whole-genome shotgun (1.7 Mb) (*r*) is shown, where the *x*-axis represents the motif size found in the input sequence and the *y*-axis represents the *n*-logarithm of the repetition number.

of both experimental phases, and according to the analysis of the distribution plots, short perfect motifs were the most frequently observed. Nevertheless, the occurrence of long motifs ensured that they did not appear by chance.[17]

Thus, in rice chromosome 10, we found examples of motifs within motifs and motifs flanked by other motifs. In the first case, one of the repetitive sequences corresponds to the LTRs of the complete retrotransposon RLX_59520 from *O. sativa* (e-value = 0.0, identity = 95%, coverage = 100%), whereas the flanking repetitive sequences have identity with the LTRs of the retrotransposon RLG_60224 from *O. sativa* (e-value = 0.0, identity = 94%; Fig. 4A).

In the second case, motifs flanked by other motifs, the inner sequence corresponds to a complete retrotransposon RLG_58802 from *O. sativa* (e-value = 0.0, identity = 89%, coverage = 99%), flanked by two complete LTRs retrotransposon RLG_5282 from *O. sativa* (e-value = 0.0, identity = 94%, coverage = 100%; Fig. 4B).

Application example 2. In this example, 107 whole-genome shotgun sequences of *Ae. tauschii*, ranging from 0.6 to 31 kb, were taken at random, totaling 1.7 Mb[4] (GenBank project AOCO000000000). This sample was selected because it contains different sequence sizes, making it useful to demonstrate the capability of the Mamushka method to identify perfect motifs nested within a certain distance, negating the problems associated with the length of the sequences.

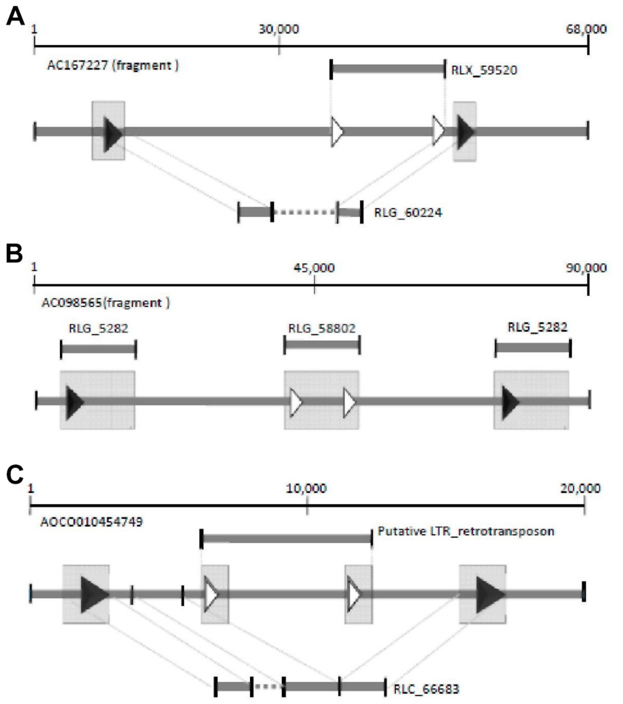


Figure 4. Nested long terminal repeat (LTR) retrotransposons found using our Mamushka approach in whole-genome shotgun sequences. Mamushka allowed the identification of perfect motifs (white triangles) flanked by other perfect motifs (gray triangles). The BLAST analysis revealed that the identified sequences correspond to LTRs (light gray boxes). (**A**) The inner sequence showed identity to the LTR retrotransposon RLX_59520, flanked by the LTRs of the LTR retrotransposon RLG_60224. (**B**) The inner sequence showed identity to the LTR retrotransposon RLG_58802, flanked at both sides by two complete copies of the LTR retrotransposon RLG_5282. (**C**) The inner sequence showed identity to a putative LTR retrotransposon flanked by two halves of another retrotransposon, RLC66683.

The distribution of perfect motifs showed a significant number of motifs with lengths ranging from 10 to 17 bp (Fig. 3). For *Ae. tauschii*, a higher frequency of motifs was detected between 5 and 60 bp as compared to longer motifs. After the fulfillment of both experimental phases, and according to the analysis of the distribution plots, short perfect motifs were the most frequently observed. Nevertheless, the occurrence of long motifs ensured that they did not appear by chance.[17]

A sequence corresponding to a putative LTR retrotransposon flanked by another LTR retrotransposon in the whole-genome shotgun AOCO010454749 from *Ae. tauschii* (Fig. 4C) was identified. In this case, Mamushka detected a pair of perfect motifs that enclosed another pair of perfect motifs, previously called "motifs within motifs". After performing an alignment using Blast-X,[18] the inner sequence showed identity to a putative gag–pol polyprotein (AAG13508; e-value = 0.0, identity = 50%, positives = 65%, protein coverage = 92%). Interestingly, the sequence encoding this polyprotein was flanked at the 5′ and 3′ end by the first (~3,800 bp) and the last region (~1,400 bp), respectively, of a Ty-1 Copia class TE (RLC_66683), suggestive of an LTR being inserted into a Copia LTR (Fig. 4C).

Matching regions are indicated by dotted lines, and the regions without a match are indicated by dashed lines in Figure 4.

These results were validated using RepeatMasker,[6,19] which identified TEs by comparing them against known elements found in a database. The positions found by Mamushka constituted a subsequence of a sequence identified by the RepeatMasker.

Conclusion

We have developed a new method, termed Mamushka, for the de novo discovery of nested motifs. Our approach identifies repetitive patterns in DNA sequences that can constitute putative nested TEs, thus simplifying the process of detecting these elements. Mamushka recovers the nested patterns in three main steps. First, it performs a search for all perfect repeats in the genome using a suffix array construction. Next, all perfect repeats are ordered according to decreasing length. Then, two pattern recognition algorithms, MWM and MFM, are executed. For MWM, given two motifs, Mamushka determines whether one motif is perfectly contained inside the other motif. The MFM algorithm searches for motifs flanked by other motifs.

The Mamushka method is exclusively focused on the identification of perfect nested patterns. Nevertheless, this feature does not constitute a drawback for the applicability of our strategy with real biological data. Any mutated TE has a conserved subsequence up to the point where the mutation occurred, allowing Mamushka to identify it. Moreover, the search does not need to be conducted in both directions because Mamushka searches for a perfect repeat. These arguments are supported by the results obtained for two different plant species, *O. sativa* and *Ae. tauschii*, in which the experiments demonstrated that our method can solve the complex problem of discovering nested TEs in actual DNA sequences.

This pattern-matching approach can be combined with existing techniques to allow for the identification of nested repetitive motifs that cannot be found with existing methodologies. We intend to test the performance of Mamushka in other plant species in future work.

Acknowledgments

We also acknowledge Dr. Carlos Lorenzetti for his technical help in the development of the Mamushka tool website.

Author Contributions

Conceived and designed the experiments: JRR. Analyzed the data: JRR, IG. Wrote the first draft of the manuscript: JRR. Contributed to the writing of the manuscript: JRR, JAC, IG, VCE, IP. Agree with manuscript results and conclusions: JRR, JAC, IG, VCE, IP. Jointly developed the structure and arguments for the paper: JRR, JAC, IG, VCE, IP. Made critical revisions and approved final version: JRR, JAC, IG, VCE, IP. All authors reviewed and approved of the final manuscript.

REFERENCES

1. Rebollo R, Romanish MT, Mager DL. Transposable elements: an abundant and natural source of regulatory sequences for host genes. *Annu Rev Genet*. 2012;46:21–42.
2. Wicker T, Sabot F, Hua-Van A, et al. A unified classification system for eukaryotic transposable elements. *Nat Rev Genet*. 2007;8(12):973–82.
3. Bennetzen JL. Transposable elements, gene creation and genome rearrangement in flowering plants. *Curr Opin Genet Dev*. 2005;15:621–7.
4. Jia J, Zhao S, Kong X, et al. Aegilops tauschii draft genome sequence reveals a gene repertoire for wheat adaptation. *Nature*. 2013;496:91–5.
5. Project IRGSP. The map-based sequence of the rice genome. *Nature*. 2005;436:793–800.
6. Bergman CM, Quesneville H. Discovering and detecting transposable elements in genome sequences. *Brief Bioinform*. 2007;8(6):382–92.
7. Kidwell MG, Lisch DR. Transposable elements and host genome evolution. *Trends Ecol Evol*. 2000;15(3):95–9.
8. Pop M, Salzberg SL, Shumway M. Genome sequence assembly: algorithms and issues. *IEEE Comput*. 2002;35:47–54.
9. Kalyanaraman A, Aluru S. Efficient algorithms and software for detection of full-length LTR retrotransposons. *Proceedings of the IEEE Computational Systems Bioinformics Conference*. IEEE, Stanford, CA, USA; August 8–11, 2005:56–64.
10. Altschul SF, Madden TL, Schäffer AA, et al. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*. 1997;25(17):3389–402.
11. Caspi A, Pachter L. In identification of transposable elements using multiple alignments of related genomes. *Genome Res*. 2006;16:260–70.
12. Lerat E. Identifying repeats and transposable elements in sequenced genomes: how to find your way through the dense forest of programs. *Heredity*. 2010;104(6):520–33.
13. Becher V, Deymonnaz A, Heiber P. Efficient computation of all perfect repeats in genomic sequences of up to half a gigabyte, with a case study on the human genome. *Bioinformatics*. 2009;25(14):1746–53.
14. Manber U, Myers G. Suffix arrays: a new method for on-line string searches. *SIAM J Comput*. 1993;22:935–48.
15. Aho AV. *The Awk Programming Language*. Addison-Wesley; 1998.
16. Cotton R. *Learning R*. O-Reilly Media Inc.; 2013.
17. Karlin S, Altschul SF. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc Natl Acad Sci U S A*. 1990;87(6):2264–8.
18. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol*. 1990;215:403–10.
19. Saha S, Bridges S, Magbanua ZV, Peterson DG. Empirical comparison of ab initio repeat finding programs. *Nucleic Acids Res*. 2008;36(7):2284–94.