TandemGraph: A Graphical Tool for Modeling String Regularities

Dina Sokol* Ramin Rakhamimov†
sokol@sci.brooklyn.cuny.edu ramin32@gmail.com
Department of Computer and Information Science, Brooklyn College of the City University of New York
2900 Bedford Avenue, Brooklyn, N.Y. 11210, Phone:(718)951-5000 ext.2065, Fax: (718)951-4842.

Submitted to: *BIOCOMP'09 - The 2009 International Conference on Bioinformatics and Computational Biology*

Abstract

In this paper we describe a software tool that we have developed to graphically depict the tandem repeats occurring in a sequence. The key advantage of this tool is that while it provides much detail, it scales well to large biological data sets. We show how this tool is used to model the tandem repeats in the human genome. Information about each chromosome that previously was displayed to a user as hundreds of pages of text can now be graphically depicted in a *single* interactive image. It possible to zoom in to view the actual sequence alignments of the tandem repeats occurring in the DNA sequence.

Keywords: *Graphical visualization, pygram, software tool, tandem repeats*

1 Introduction

In the past several years, there has been an explosion in the amount of the stored biological data. With the complete sequencing of the human genome, and the computational analysis done on the sequences, there are massive datasets available for study. This presents new challenges for researchers in computational biology. In order to maximize the use of this data, it is imperative that the data be organized and presented in a way that facilitates the discovery of new knowledge and the advancement of the field of biology.

One important aspect of the analysis of the genomes of eukaryotic and prokaryotic organisms, is the location of repeated sequences within the genome. Repeated sequences comprise 50% of the human genome, and a significant percent of the genomes of several other organisms. *Tandem* repeats are sequences that are repeated two or more times contiguously. Tandem repeats in DNA, also called satellite DNA, are important in numerous fields including disease diagnosis, mapping studies, human identity testing, sequence homology, and population studies [5, 6, 11].

Sokol et al. [10] have recently developed TRED - software for locating Tandem Repeats over the Edit Distance within a DNA sequence. Other software tools for finding tandem repeats in a sequence include: TRF (Tandem Repeats Finder) [1], mreps [7], ATRHunter [12], STRING [9] and TandemSWAN [2]. Each of these tools implements an original algorithm to detect tandem repeats within a sequence, under its own definition of approximate tandem repeats. Common to all tools is the finding of an abundance of tandem repeats within the human genome. For example, in Chromosome 1 of Homo Sapiens, TRF locates 72,530 repeats, and TRED locates 91,815 repeats. This is after heuristics are applied to combine and filter the

*This work has been supported in part by the PSC-CUNY Research Award 69273-0038.
†This work has been supported in part by the National Science Foundation Grant DB&I 0542751.

Alignment	Start	End	Length	Period	Repetitions	Errors	Score
View	1	468	468	6.1	77.2	20	382
View	621	860	240	74.8	3.2	7	140
View	9,169	9,308	140	70	2	5	50
View	20,718	20,755	38	1.9	20	4	20
View	20,726	20,785	60	13	4.6	6	24

Table 1: The first 5 lines of the table of repeats found in Chromosome 1 of Homo Sapiens.

repeats that are found initially. Currently, these repeats are displayed to the end-user in a table, each line containing statistics about one tandem repeat (see table 1 for e.g.). We paginate the table, displaying 100 repeats per page, yielding close to 1,000 pages per chromosome. If we want biologists to be able to analyze this data to advance computational genomics, it must be presented in a clear graphical visualization, allowing both a high level overview, and variant levels of detail.

There have been some attempts at designing a GUI interface to represent repeats within an entire chromosome. The JSTRING [9] software has a graphical view of its results. We found that its interface is not very intuitive. The pygram model [4] is very intuitive, yet it works for exact repeats (i.e. without errors) that are not necessarily tandem. Reputer [8] also provides a graphical representation of pairwise occurrences of non-tandem repeats, yet it does not scale well to large sequences.

In this paper we describe a software tool called TandemGraph that we have developed to graphically depict the tandem repeats in a sequence. The idea of the representation is largely based upon the model of the pygram (or pyramid diagram) [4], which uses overlaid triangles, in a similar manner to an earlier design called the *landscape* [3]. Our representation allows one to view the entire set of tandem repeats in a chromosome in a single image, and then to continuously zoom all the way down to viewing the actual sequence of bases included in each repeat. In the next section, we describe the graphical model that we have designed. In the Results section, we show how the tool depicts the repeats of several of the chromosomes in the human genome. We conclude with a discussion of how this model can be used for all of the software tools mentioned above, as well as to depict other types of regularities in a sequence.

2 Methods

2.1 Graphical Model

Our model uses overlaid colored triangles to represent the substrings of the input string that have been reported as tandem repeats (see Figure 1). Each triangle represents one tandem repeat (which by definition can be split into approximate periods). Given a sequence S of length n, and a list of the substrings of S that are tandem repeats, a representation is a 2-dimensional graph, where the x-axis is labeled with the actual sequence, and triangles are drawn in the matrix above, with the height of each triangle representing the length of the repeat. The left x-coordinate of each triangle represents the first nucleotide of the repeat sequence, while the right x-coordinate represents the end of the repeat.

Formally, given a value *xScale* and *yScale*, each representing the scale factor of the x and y axis, respectively, the ith nucleotide of S will be at location $i/xScale$. Each repeat from beginning position *begPos* to end position *endPos* of length ℓ, is drawn as an isosceles triangle with points $begPos/xScale$, $endPos/xScale$, and $\ell/yScale$. In practice, the values of *xScale* and *yScale* are calculated from the width and height of the current window, which is resizable by the user. For the exact formulas that calculate the points of each triangle, see the java code that appears in Figure 3.

Remark: It is important to note that the multiple alignment of the periods of a given approximate tandem

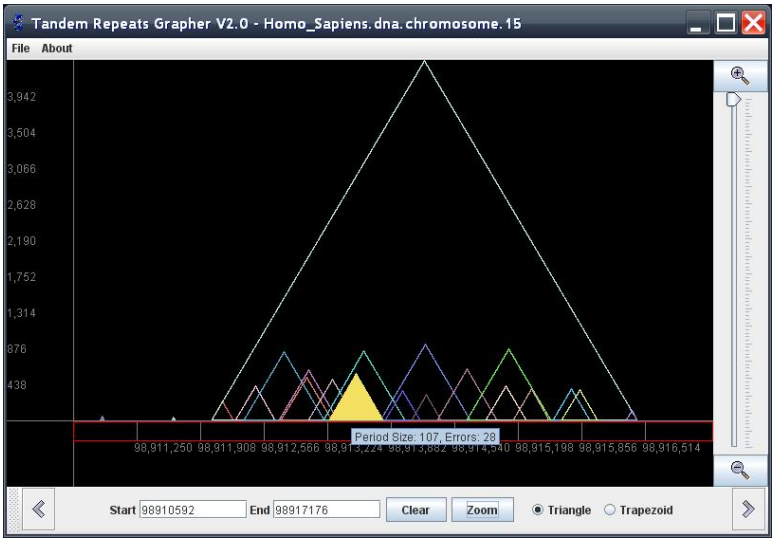


Figure 1: This graph shows how overlaid colored triangles are used to represent tandem repeats.

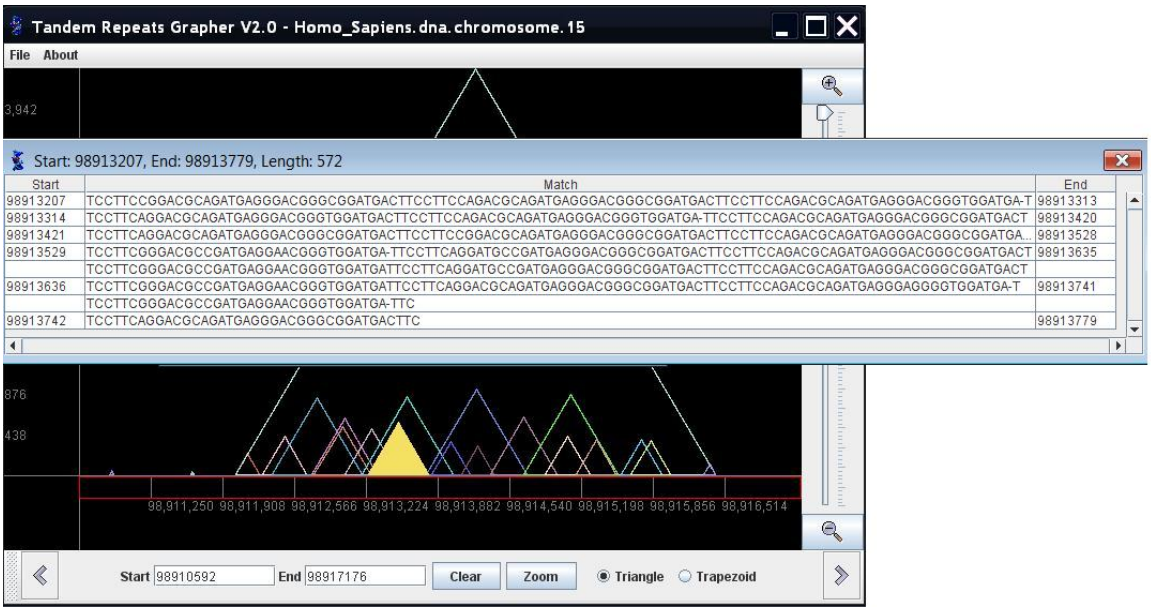


Figure 2: When a triangle is clicked, the multiple alignment of the repeat's periods is displayed in a pop-up textbox.

$ScalesMethods:$
$xScale = (double)repeats.getRange()/(component.getWidth() - LEFT\_MARGIN);$
$yScale = (double)repeats.getMaxSize()/(component.getHeight() - BOTTOM\_MARGIN);$

$TranslationMethods:$
$privateinttranslateXPoint(intx)\{$
$return(int)((x - repeats.getStartPoint())/xScale) + LEFT\_MARGIN;$
$\}$
$privateinttranslateYPoint(inty)\{$
$returncomponent.getHeight() - BOTTOM\_MARGIN - (int)(y/yScale);$
$\}$

$ZoomMethod:$
$intactualX = (int)((x - LEFT\_MARGIN) * xScale) + repeats.getStartPoint();$
$updateVals(actualX - (repeats.getRange()/ZOOM\_LEVEL/2), actualX + repeats.getRange()/ZOOM\_LEVEL/2));$

Figure 3: The formulas that calculate the three points of each triangle that represents a tandem repeat.

repeat is often *not unique*. Due to the possible insertions and deletions in the copies of a repeat, it is often possible to break up the repeat into approximate periods, in several different ways. Thus, it is appropriate that each triangle represent the entire repeat, as opposed to the exact case of the pygram, where each *period* is represented by its own triangle.

The triangles are outlined, therefore all overlapping repeats are clearly visualized. Information about other attributes, such as period size and percent error, appear in a triangle as the user mouses-over the triangle. In addition, as the mouse is placed in a triangle, the triangle gets filled in, to clarify which of the overlapping triangles is selected. Once a triangle is selected, the user can click on it to view the multi-alignment of the actual repeat (see Figure 2).

Zooming Features: The highest level view represents an entire chromosome. For this level the graph generally degenerates to a column graph, each column representing a repeat, with the height of the line representing the repeat's length. If there is more than one repeat located at the same (or close) location, the column will appear as a multicolored line, the bottom part representing the shortest repeat, the piece above another color representing the next longer repeat, and so on.

Zooming has been implemented in several different user-friendly ways. The slider at the right provides a continuous zoom, with zoom-out and zoom-in buttons on top and bottom. The red rectangle on the bottom allows the user to drag a range of the chromosome to zoom, while the text boxes underneath the rectangle allow the user to enter actual start and end locations in the chromosome. All three of these zooming features are fully integrated, so that the actual indices appear in the start and end box as the user drags through a range.

2.2 Log-graph

In some chromosomes there are repeats that are extremely long, possibly spanning over 100,000 bases. These long repeats cause the short tandem repeats (which are much more common) to be barely visible as their heights scale to close to zero. In situations where data covers a large range of values it is common to present the data on a logarithmic scale. This has been done in the pygram, which translated to a log-pygram, and we offer this option as well. Thus, the *y-axis* is labeled with powers of 10 (10, 100, 1000, etc). A repeat with length ℓ has height $\log_{10} \ell$. Using the log-scale, a triangle that represents a repeat that is a substring of another repeat will not necessarily fit entirely inside its superstring's triangle. Therefore, we represent repeats as trapazoids. The left and right x values are the same as described above for the triangles, and the height of the trapazoid is $y = \log_{10} \ell$ where ℓ represents the repeat's actual length. TandemGraph includes radio buttons to allow the user to switch the graph from triangles and linear heights to trapazoids and log-scale heights.

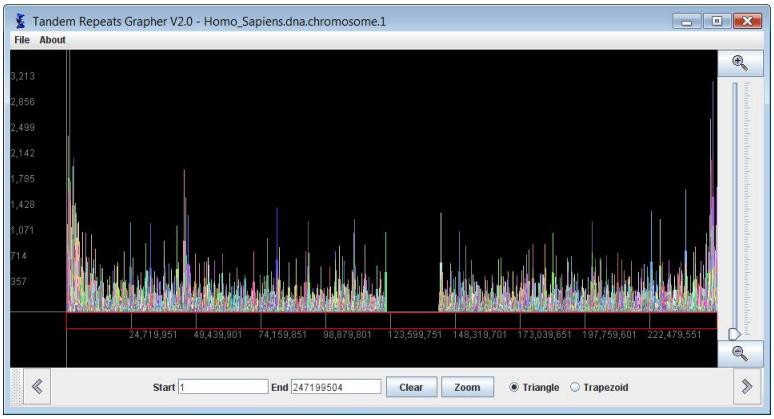


Figure 4: This graph depicts all of the tandem repeats found by TRED in chromosome 1 of homo sapiens. Each vertical line represents a tandem repeat.

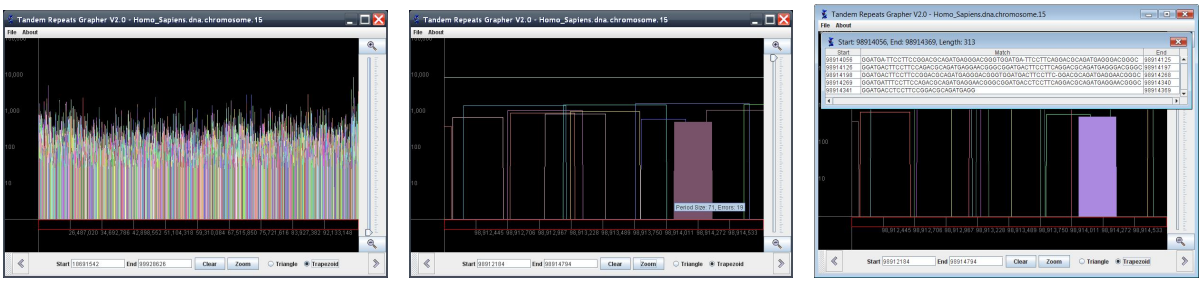
2.3 Implementation Details

Language: The core program has been written in Java. Swing was used to provide the necessary graphical components, and the Java AWT was used for event handling. The project was developed in the Eclipse IDE, which provided a friendly environment for students, and allowed easy sharing of the modularized package. The following open source technologies were used as well: *MySql DBMS* was used for the database, *Spring* was used to simplify the database access routines and provide an elegant method of implementing inversion of control. *Log4j* permitted us to have a more fine-tuned control over debugging the different states and procedures of the application during run-time. The *commons-dbcp* package was used for adding database pooling functionality. *JUnit* was used for conveniently testing the separate modules; *Maven* was used as the build tool, and *git* was used as the version control system.
Model-View-Controller (MVC): The Model-View-Controller is a popular software design paradigm. The *model* refers to the code that deals with the data, the *view* refers to the code that manages the user-interface, and the *controller* controls the flow of the entire application using interfaces to the models and the views. The idea is to allow changes to be made either to the underlying data or the visual appearance of the application, without affecting the other component. TandemGraph was developed using MVC, this enabled an intuitive and simplified approach for the development process, and for future changes.

3 Results

We have run TandemGraph on all 23 chromosomes of Homo Sapiens and the results are excellent. TandemGraph provides a GUI interface to huge amounts of data, previously available as text only. In Figure 4 we show the graph for all 91,815 repeats found in chromosome 1. This figure represents information that previously had to be viewed on 918 different pages.

The data for all 23 chromosomes is available online at the TRED website. In fact, we have fully integrated the tool with the database, and provided a link from the TRED website[1] to TandemGraph. Thus, when the user opens the application, there will be a menu of all of the chromosomes in the human genome. When a chromosome is chosen, the application automatically connects to the TRED database, downloads



(a) This graph shows the repeats found in the entire chromosome 15.

(b) The graph of chromosome 15 is zoomed in to display index 98,912,184 to index 98,914,794.

(c) A trapazoid has been selected, and the multi-alignment of the selected repeat is displayed.

Figure 5: Log-graph depicting the tandem repeats in Homo Sapiens chromosome 15.

the information about the repeats in that particular chromosome, and draws the graph.

In Figure 5 we show the log-graph of chromosome 15 of the homo sapiens. The longest repeat found by TRED in chromosome 15 has length 4,388. The highest level graph is shown (a), as well as two zoomed in graphs (b and c).

4 Discussion

We have developed an open source software tool, TandemGraph, that graphically depicts the set of tandem repeats found in a genome. The software is very simple and easy to use. We have shown how TandemGraph works in conjunction with TRED, a tandem repeats software tool. In addition, TandemGraph can be coupled with other tandem repeats software tools (such as those mentioned in Section 1).

The basic concept of drawing triangles to represent a repeat comes from the pygram. We have essentially constructed a *generic pygram*, which can be used to represent tandem repeats under any definition of a tandem repeat. In particular, tandem repeats usually contain insertions, deletions, and replacements of nucleotide sequences. Furthermore, our new representation can be used to represent other regularities in a sequence such as the set of inverted repeats, palindromes, or any other set of significant substrings of a sequence. TandemGraph can accept as input any set of substrings that is arranged as a listing of starting and ending locations within a sequence. We believe that this tool will prove to be very useful in many existing and future applications.

Acknowledgments

The authors would like to thank Professor Shaneen Singh of the Biology Department at Brooklyn College for contributing a biologist's perspective to this project, and Professor Gerald Weiss for fruitful discussions about the software development.

[1] http://tandem.sci.brooklyn.cuny.edu

References

[1] G. Benson. Tandem repeats finder – a program to analyze DNA sequences. *Nucleic Acids Research*, 27:573–580, 1999.

[2] V. Boeva, V. Makeev, and M. Régnier. SWAN: searching for highly divergent tandem repeats in DNA sequences and statistical significance. In *JOBIM'04*. IEEE Computer Society, 2004. In Proceedings JOBIM'04, Montréal.

[3] B. Clift, David Haussler, Ross M. McConnell, Thomas D. Schneider, and Gary D. Stormo. Sequence landscapes. *Nucleic Acids Research*, 14(1):141–158, 1986.

[4] Patrick Durand, Frédéric Mahé, Anne-Sophie Valin, and Jacque Nicolas. Browsing repeats in genomes: Pygram and an application to non-coding region analysis. *BMC Bioinformatics*, 7:477, 2006.

[5] C. T. Caskey et al. An unstable triplet repeat in a gene related to Myotonic Dystrophy. *Science*, 255:1256–1258, 1992.

[6] A. J. Jeffreys. DNA typing: approaches and applications. *Journal of the Forensic Science Society 33*, pages 204–211, 1993.

[7] R. Kolpakov and G. Kucherov. mreps: efficient and flexible detection of tandem repeats in DNA. *Nucleic Acids Research*, 31:3672–3678, 2003. http://www.loria.fr/mreps/.

[8] Stefan Kurtz and Chris Schleiermacher. Reputer: fast computation of maximal repeats in complete genomes. *Bioinformatics*, 15(5):426–427, 1999.

[9] Valerio Parisi, Valeria De Fonzo, and Filippo Aluffi-Pentini. String: finding tandem repeats in dna sequences. *Bioinformatics*, 19(14):1733–1738, 2003. http://bioinf.dms.med.uniroma1.it/JSTRING.

[10] D. Sokol, G. Benson, and J. Tojeira. Tandem repeats over the edit distance. In *Bioinformatics*, volume 23, pages e30–e35, 2006. http://tandem.sci.brooklyn.cuny.edu/Tandem.

[11] M.W. Uform and R.K. Wayne. Microsatellites and their application to population genetic studies. *Current Opinion in Genetics and Development*, 3:939–943, 1993.

[12] Ydo Wexler, Zohar Yakhini, Yechezkel Kashi, and Dan Geiger. Finding approximate tandem repeats in genomic sequences. In Philip E. Bourne and Dan Gusfield, editors, *RECOMB*, pages 223–232. ACM, 2004.