TandemGraph - A Tool for Modeling String Regularities

Ramin Rakhamimov
ramin32@gmail.com

Under the Supervision of Associate Professor Dina Sokol
sokol@sci.brooklyn.cuny.edu

CIS 790 - Research Project

Department of Computer and Information Science,
Brooklyn College of the City University of New York
2900 Bedford Avenue, Brooklyn, N.Y. 11210

24th December 2009

Abstract

We have developed a graphical tool to model regularities in biological sequences. TandemGraph enables a user to study and analyze string regularities. An important application of this tool is when used to model tandem repeats found within chromosomes. We have used TandemGraph to graphically represent the tandem repeats in the human genome found by a tandem repeats program called TRed. The tool is available online and the source code is freely distributed upon request.

1 Introduction

Most genomes have high content of tandem repeats, and the homo sapiens is no exception. In the sequence of chromosome 1, our tandem repeats program, TRed [3], locates 91,815 repeats. In the table view of the database, these repeats are displayed to the end-user in tables, one line per tandem repeat, 100 lines per table. This yields 919 pages for chromosome 1 alone. In order for biologists to be able to analyze this data, we felt that it must be presented in a clear graphical visualization, allowing both a high level overview, and variant levels of detail. To this end, we have developed a new software tool called *TandemGraph* to graphically depict the tandem repeats in a sequence. TandemGraph allows one to view the entire set of tandem repeats in a chromosome in a single image, and then to continuously zoom in to see further details.

We have run TandemGraph on all 24 chromosomes of Homo Sapiens (1-22,X,Y) and the results are excellent. TandemGraph provides a GUI interface to huge amounts of data, previously available as text only. We have fully integrated the TandemGraph tool with the TRedD database. Thus, when the user chooses “View Graph," the TandemGraph application opens and automatically connects to the TRedD database, downloads the information about the repeats in that particular chromosome, and draws the graph. Furthermore, there is a menu of all of the chromosomes in the human genome in TandemGraph, so that the user can switch to a different chromosome without returning to the browser.

Using TandemGraph, the repeats are drawn on a 2 dimensional plane, i.e. X is the position within the original pre-filtered input, Y is the length of the repeat. Each repeat is depicted as a triangle. The base of the triangle, together with the height, both represent the length of any given repeat.

2 Features

Once TandemGraph is loaded, the following options are available to the user:

- Load any of the pre-filtered inputs from our in-house database.
- Input can be zoomed in or zoomed out.
- Input can navigated to the left or the right.
- Lookup repeat lengths and number of errors found.
- Switch between triangle and trapezoidal views.
- Bring up the repeat alignment on demand for any individual repeat.

2.1 Loading Inputs

Upon start, TandemGraph dynamically looks up the latest chromosome index in the TRedD database, and populates them within a file menu. The user is then required to pick a desired chromosome as an input to begin analysis. The user has an option of switching between chromosomes within the same session. Inputs can also be cleared to begin new sessions.

2.2 Navigation

After a chromosome has been successfully loaded into the application a user has several options, with which to proceed. The 2 most important navigation options are zooming and shifting. For zooming, the user may enter a specific region that he/she would like to zoom in on in a text box field. Positioned below the repeats is a rectangular area designed to zoom in on regions by dragging a desired region using the naked eye. There is also a scrollbar that enables the user to continually zoom in or out with respect to the center of the current display. Along with the scrollbar there are 2 buttons (iconified as magnifying glasses) that enable the zooming to take place at discrete intervals. When the region that is being currently displayed is equivalent in size to the width of the window, the original pre-filtered chromosome data is displayed below the view.

The user may also shift the current display to the right or to the left. Shifting is done via the use of left and right navigation buttons found at the bottom of the application window. On each shift the current display is shifted in the right or left direction by half the size of the currently selected region.

At times the repeat length of any particular input may vary by a large amount. When this occurs, smaller repeats are not easily discernible to the naked eye. We have adapted the usage of logarithmic approximation as done in Pygram [4]. By applying the function:

$$f(x) = \log_{10}(x)$$

to the length of all the repeats, we are able to maintain differences in length while averaging them out at the same time.

2.3 Statistics

If an individual repeat is discernible within the current view it may be used to bring up various statistics about itself. By hovering over the repeat with the mouse cursor, the repeat gets highlighted with its own predefined color. Once a repeat is highlighted a tool-tip is displayed with the period size and the number of errors found in calculating the approximate tandem repeat [3]. A highlighted triangle may be clicked on, to bring up its alignment. The alignment will the matching part of the original input along with the error positions and repeat sizes.

3 Design

3.1 Code Organization

Due to the complexity of the application, a strategy was required in simplifying the organization of the code. By separating the views, business logic and state, the Model-View-Controller [5] (MVC) design pattern provided us with an efficient method of managing the application code.

3.2 Implementation Details

All scaling routines are implemented via the use of the following two functions:

$$\begin{aligned} translateXPoint(x, startPoint, xScale) &= ((x - startPoint)/xScale) \\ translateYPoint(y, yScale, gridHeight) &= (gridHeight - (y/yScale)) \end{aligned}$$

Where the scales are initialized as follows:

$$\begin{aligned} yScale &= maxRepeatLength/currentGridHeight \\ xScale &= currentRegionWidth/currentGridWidth \end{aligned}$$

The individual repeats are redrawn as follows.

$$\begin{aligned} &actualX = ((x - leftMargin) * xScale) + repeats.getStartPoint() \\ &updateVals(actualX(repeats.getRange()/zoomLevel/2), \\ &\qquad actualX + repeats.getRange()/zoomLevel/2)) \end{aligned}$$

The vertex of each triangle is simply the midpoint of the base.

3.3 Technology

The project has been written using the Java Programming Language. For 2D graphics and even handling we employed the Swing Library (part of the Java API). The development environments used were Eclipse, IDEA, and

GNU Vim. GNU Linux was used as the operating system of choice for all development on this project. The MySQL Database

The following open source technologies were used:

- MySQL DBMS was used for the database.
- The commons-dbcp package was used for database pooling functionality.
- Spring Framework was used to simplify database access and provided 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. JUnit was used for conveniently testing the separate modules;
- The ant build tool provided us with a comprehensive build tool.
- For version control we chose to use git, due to its easy and efficient interface.

4 Acknowledgments

Professor Shaneen Singh and Professor Gerald Weiss deserve our thanks for their various contributions in the development of the project.

References

[1] G. Kucherov and D. Sokol. *Approximate Tandem Repeats.* Encyclopedia of Algorithms, 2008.

[2] G. M. Landau, J. P. Schmidt and D. Sokol. *An Algorithm for Approximate Tandem Repeats.* Journal of Computational Biology, Volume 8, p. 1-18, 2001.

[3] D. Sokol, G. Benson, and J. Tojeira. *Tandem Repeats over the Edit Distance.* Bioinformatics 2007 23(2): e30-e35

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

[5] Steve Burbeck, Ph.D., *How to use Model-View-Controller (MVC).* Applications Programming in Smalltalk-80(TM).