PQuad: Visualization of Predicted Peptides and Proteins

Susan L. Havre[1] Mudita Singhal[2] Deborah A. Payne[3] Bobbie-Jo M. Webb-Robertson[4]
Pacific Northwest National Laboratory

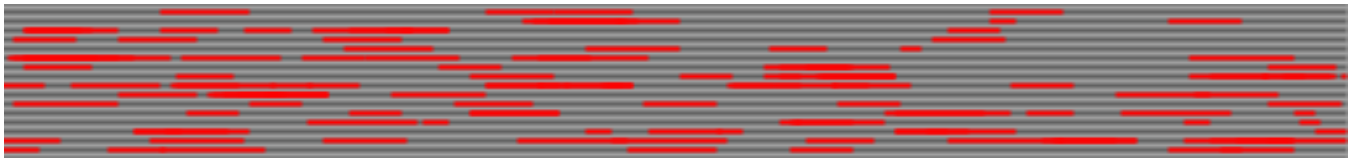

Figure 1: Peptides identified from mass spectrometry data on a protein sample from a controlled experiment**.** Peptides are shown in red in their correct locations along a DNA segment represented by a wrapped dark gray line. Areas with overlapping peptides appear slightly brighter red. Among other things, biologists hope to infer protein function from the peptide composition of biological samples prepared under different experiment conditions, for example, at different times during a cell's development cycle or under varying temperatures or oxygen levels.

ABSTRACT

New high-throughput proteomic techniques generate data faster than biologists can analyze it. Hidden within this massive and complex data are answers to basic questions about how cells function. The data afford an opportunity to take a global or systems approach studying whole proteomes comprising all the proteins in an organism. However, the tremendous size and complexity of the high-throughput data make it difficult to process and interpret. Existing tools for studying a few proteins at a time are not suitable for global analysis. Visualization provides powerful analysis capabilities for enormous, complex data at multiple resolutions. We developed a novel interactive visualization tool, PQuad, for the visual analysis of proteins and peptides identified from high-throughput data on biological samples. PQuad depicts the peptides in the context of their source protein and DNA, thereby integrating proteomic and genomic information. A wrapped line metaphor is applied across key resolutions of the data, from a compressed view of an entire chromosome to the actual nucleotide sequence. PQuad provides a difference visualization for comparing peptides from samples prepared under different experimental conditions. We describe the requirements for such a visual analysis tool, the design decisions, and the novel aspects of PQuad.

CR Categories: I.3 Computer Graphics, J.3 Life and Medical Sciences

Keywords: visualization, metaphor, context, proteomics, differential proteomics, difference visualization

1 INTRODUCTION

The Human Genome Project brought about major advances in genomics. Sequencing a genome, the information storage unit of an organism, is now primarily a matter of selecting the organism and having the necessary equipment, skills, and time. Proteomics is the new big challenge [1]. Proteins are the cell's mechanism for putting an organism's genomic information into action. A proteome is the collection of all proteins present in an organism. Unlike the genome, the proteome is dynamic, changing continuously in response to tens of thousands of intra- and extra-cellular environmental signals. The proteome is an essential key to understanding the complex processes of cells. *Which proteins are present*, *when and where are they present, what state are they in, and what is their function* are the crucial questions in proteomics research. The success of proteomics will rely on high-throughput experimental techniques coupled with sophisticated data analysis methodologies.

Mass spectrometry (MS) is at the cutting edge of proteomic technologies. High-throughput MS provides extremely precise mass measurements of thousands of proteins or peptides (protein fragments) in a biological sample from a single experiment. The voluminous raw MS data contains evidence of the proteins present in the sample. Valuable information such as protein identity, quantity, interactions, and modifications can be inferred from this evidence. However, the MS (mass) data must first be mapped to protein sequences. Typically, proteins are cleaved by enzymatic digestion into peptides prior to the MS analysis. The peptide MS masses are then mapped to peptide and, finally, protein sequences.

Typically, peptide identification software is used for mapping MS data [2-4]. More accurately, current software *predicts* peptide identity from the MS data. Such software produces a list of identified peptides with each peptide's sequence, the source protein or proteins, and metrics produced during the identification process. Further analysis is required to validate the identification and progress from peptide identification to protein identification and on to understanding the proteome. Even when the number of resulting peptide identifications is small, the subsequent analysis and information extraction is time-consuming and challenging. As the number of identified peptides grows, navigating and analyzing a data set becomes even more challenging. Understanding the difference in peptide sets collected—for instance, during different points in the cell life cycle—is especially challenging. Nevertheless, the comparison of two or more sets of identified peptides, d*ifferential proteomics*, is a key to understanding proteins.

Biologists need powerful computational tools to assist in the analysis of large, multiple proteomic data sets. Visualization abstracts and depicts large-scale data sets in an interactive visual representation designed to ease cognitive tasks and enable the analysts to see patterns and relationships not distinguishable otherwise [5]. Currently, well-developed, powerful software tools are available for studying and analyzing *genomic* data. Some of these tools, such as GeneSpring [6], OmniViz [7], and Spotfire [8], support the visual

[1] susan.havre@pnl.gov
[2] mudita.singhal@pnl.gov
[3] debbie.payne@pnl.gov
[4] bobbie-jo.webb-robertson@pnl.gov

IEEE Visualization 2004
October 10-15, Austin, Texas, USA
0-7803-8788-0/04/$20.00 ©2004 IEEE

analysis of experimental data, for example, gene expression data from microarray analysis. Similar powerful visualization tools do not exist for the visual analysis of experimental proteomics data. This paper describes a visualization tool we developed to support analysis of identified MS peptides and proteins. Figure 1provides a preview of the tool's visualization of peptides identified from experimental biological samples.

1.1 Biological Terms

The following is a simplified introduction to the biological terms used in this paper. All the information for an organism is stored in its genome as one or more units (**chromosomes** or **plasmids**) of deoxyribonucleic acid (**DNA**). Each unit has two strands that form a double-helix molecule. Each strand is a sequence of connected **nucleotides**, either adenine (A), thymine (T), cytosine (C), or guanine (G). The genomic sequence is specified by only one strand. The sequence of the second strand can be inferred from the first because the strands are linked by **basepairs**, or complementary pairs, A-T and C-G. For example, wherever there is an A in the primary strand, there is a T in the complement strand. Each strand has distinguishable ends referred to as the 5' and the 3' end. The 5' end of one strand is linked to the 3' end of the other strand. Each strand is decrypted from the 5' end to the 3' end, that is, they are read in opposite directions as indicated in Figure 1. Genetic information is stored in specific regions of the genome; the regions of primary interest in this paper are **genes** that translate to proteins. A gene segment is highlighted in yellow along the 5' to 3' DNA strand in Figure 2.

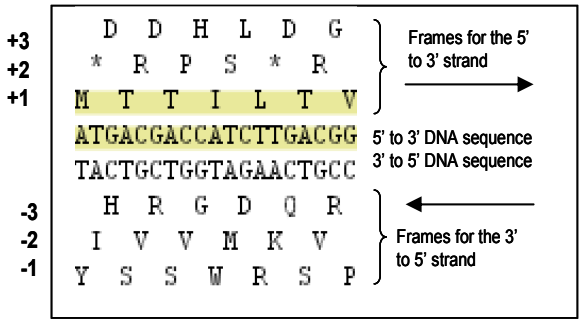

Figure 2: Sample section of DNA strand sequences with their associated frame sequences. Frames labels are shown on the left.

Proteins are large, complex molecules of amino acids, which are translated from a gene's nucleotide sequence in triplets, or **codons**. Each codon is translated to one of 20 possible amino acids or to a stop codon that signals the end of the gene. It is important to recognize that, given a nucleotide sequence, there are three possible amino acid sequences depending on where you start translating codons. Consider the nucleotide sequence of the 5' to 3' strand in Figure 2: ATGACGACCATCTTGACGG. If you start at the first nucleotide, you will get one set of codons: (ATG) (ACG) (ACC) (ATC)…. If you start at the second nucleotide, your codon set will be (TGA) (CGA) (CCA)…. If you start at the third nucleotide, you will get another set of codons, (GAC) (GAC) (CAT)…. The first codon, (ATG) translates to the amino acid methionine which is represented by the letter "M". The translation of these three codon sets to amino acids is MTT (methionine, threonine, threonine or MTT), *RP, and DDH, respectively. These three possible amino acid sequences are called **frames**. The complement nucleotide string also has three frames. In total, there are six possible frames as shown in Figure 2. **Open reading frames** (**ORFs**) give the amino acid sequences associated with gene nucleotide sequences. Protein sequences are either the same as the ORF sequences or modifications of them. In Figure 2, the ORF associated with the highlighted gene on the 5' to 3' strand is also highlighted in yellow along the +1 frame.

The ORF locations are predicted by a variety of methods algorithmically and experimentally. A defined set of ORFs for a genome is called the **annotation**, which can be obtained from resources such as The Institute for Genomic Research (TIGR) [9]. For MS proteomics experiments, proteins are broken into fragments called **peptides**. The amino acid sequence of a peptide matches a contiguous section of the parent protein. For additional information, see the sidebar in [10].

1.2 Traditional Genomic/Proteomic Graphics

Traditionally ORFs are depicted as lines, bars, or boxes drawn on or parallel to a horizontal line representing a DNA segment. Usually labels indicating ORF names and sequence indices are provided for context. Most computerized graphics fail to exploit the capabilities of interactive visualization. Links, if available, are typically static popup windows with no connection back to the visualization. Only a relatively small segment of a sequence is shown at once. Navigation is typically awkward, advancing in chunks at the click of a button or by entering text. Changes in resolution, if offered, only enlarge or contract the same view with no change in the level of detail. As a result, navigation is awkward, context is limited, and the visualization does not support a variety of proteomic research tasks.

2 Related Work

Jaffe et al. [11] demonstrate a method to generate an improved ORF annotation as discussed later in the paper. They created a simple, web-based visualization called Proteogenomic Map Viewer (http://massive.med.harvard.edu/cgi-pub/superviewer.cgi). The viewer graphically depicts vertically aligned blocks of ORF areas for multiple sets of predicted ORFs, the ORF set differences, and identified peptides. Sections of the genome sequence are accessible in chunks. The software handles only the target organism of the authors experiment, *Mycoplasma pneumoniae*, although the authors plan to generalize the tool to other organisms. Sequence information for the ORFs or processing information on the peptides is accessed by clicking on the ORF or peptide blocks to bring up static windows with the detail information. Proteogenomic Map Viewer is designed for analyzing experimental proteomic data, but the visualization techniques are very basic.

An excellent (and free) genome browser, Artemis [12] developed by the Sanger Center (Cambridge, UK), can be used to view peptide identification by formatting the peptide identification software results as an EMBL (European Bioinformatics Institute) [13] or GenBank [14] feature table. This is the same format used to input the protein definitions.

In comparison to the work described here, PQuad is designed specifically for the analysis of experimental peptides and proteins through interactive visualization.

3 Requirements

We identified the following requirements for a visual analysis tool for predicted peptides and proteins.

3.1 Scalability

The number and size of the chromosomes and plasmids associated with an organism vary widely. Even when limiting the field to microorganisms, the data range from thousands to hundreds of millions of nucleotides. PQuad must handle not only long nucleotide sequences, but also large numbers of ORFs (>10,000) and peptides (>100,000). The peptides and proteins may have associated information such as pedigree, sequence, or uncertainty that must be tracked.

Finally, large amounts of related biological information such as gene function or cellular location of proteins may be requested by the user; these must also be tracked.

3.2 Easy Navigation, Quick Response

The range of resolution in the data from an entire chromosome to the DNA sequence requires the ability to quickly and easily get to the desired level of resolution without losing context. The user must be able to determine the current focus location in a view as well as in linked views.

3.3 Appropriate Context

The complexity of the data requires that context information be readily available. Peptides must be seen in the context of their parent proteins as well as in the context of the DNA strand. Also, because the data could be integrated from multiple sources, users must be able to see qualitative information that identifies data sources, including the experimental data—for example, the organism, data processing information, and quantitative information such as the size of the data and the counts of peptides and proteins.

3.4 Difference Analysis

While exploring a single data set will be an important task, comparing data sets is even more important. Understanding the difference between proteins present under different conditions will provide greater insight into the role of those proteins in the cell.

3.5 Flexibility

Although the overall objectives of a research prototype remain steady, the final product is not always seen clearly from the beginning. Often, as prototype capabilities progress, one gains insight on better approaches or capabilities to investigate. Such prototypes must be built lean (not over-designed) yet flexible enough to adapt to changes in direction.

3.6 Usability

It is usually difficult to convince users to learn a new tool to do something that they have been doing for years using conventional methods, such as spreadsheets [15]. We consider it essential that the biologists and bioinformaticists see value in the prototype and be willing to use it in real analysis tasks.

4 Novel Aspects of PQuad

PQuad is an interactive visualization tool designed to survey and analyze peptide evidence from proteomic experiments.

4.1 Experimental Peptide and Protein Visualization

PQuad provides linked views of the experimental data at multiple levels of resolution and detail allowing biologists to view empirical evidence of peptides (and therefore proteins) in the context of the genome and proposed ORF annotations. PQuad provides three key resolution levels necessary for the analysis of peptides and proteins. Each resolution provides a different level of detail. The highest and lowest resolution views are fixed. The intermediate resolution supports user-control of the resolution as well as a number of display options; only one option is shown in this paper. In addition PQuad displays descriptive, quantitative, and qualitative information on the experimental data, its processing, and other pertinent details, as available.

4.2 Wrapped Line Metaphor

PQuad employs a wrapped line metaphor to represent the DNA sequence. The metaphor of a wrapped line has several advantages. First, it is an obvious extension of the traditional view where a line segment represents only a small part of the DNA sequence. Wrapped lines are a familiar concept to anyone who reads; the parallel between letters or words in a paragraph to genomic sequences represented as a continuous string of alphabetical characters (without blank spaces or punctuation) is obvious. Further, a wrapped line allows more context and information to be presented in a view than most alternatives.

4.3 Comparison of Peptide Sets (Difference Visualization)

Although analyzing a peptide set may be difficult, comparing multiple peptides sets, called *differential proteomics*, is extremely difficult. However, such comparisons may provide important insights. For example, biologists might compare peptide sets from an organism's cells prepared under controlled conditions with one variable, such as temperature or oxygen levels. The peptide sets could be compared side-by-side using two instances of PQuad, each with a different peptide set. But side-by-side comparisons become more difficult as the distance between the focal point in each view increases. PQuad provides a difference visualization that depicts differences in peptide sets in the same view.

5 Design Decisions

The key design decisions relevant to the PQuad features presented in this paper are discussed below. The fact that PQuad is an application prototype effects design decisions; we explore options and work with users for feedback seeking new functionality with the potential for significant impact in proteomics research.

5.1 Multiple Views

Talking with potential users and surveying current visualization metaphors for genomic data revealed the need for two distinct resolutions, each with a different appropriate level of detail. The resolution in Figure 3 depicts multiple, contiguous ORFs at the resolution of the traditional visualization described in Section 1.2. At this resolution, an ORF of interest can be clearly seen in the context of its immediate neighbors. This is convenient, for example, to identify proteins that work together. The resolution in Figure 4 is convenient for reading the text characters in both nucleotide (DNA) and amino acid (ORF) sequences. Finally, information visualization research, particularly on the Information Mural [16], has demonstrated the benefit, of a bird's eye view for context and navigation. We defined a third resolution giving a compressed, but global view as described in detail below.

Figure 3: Resolution and detail for viewing ORFs. The DNA strands are depicted by two black lines; the associated ORFs are shown as yellow bars overlaid with the identified experiment peptides in red. Some ORFs have no associated peptides; some have many.

Having identified three key resolutions, the next issue was how to bridge them. After considering continuous zooming, we decided multiple linked windows with preset resolutions are a better approach for the proteomic data and analysis tasks. Continuous zooming through a display of massive data over such a wide scale to find the few useful resolutions would be difficult for the user. We also considered a focus+context approach, but since many analysis tasks require simultaneous, large displays at multiple resolutions, we decided again in favor of the linked views. Multiple linked views allow

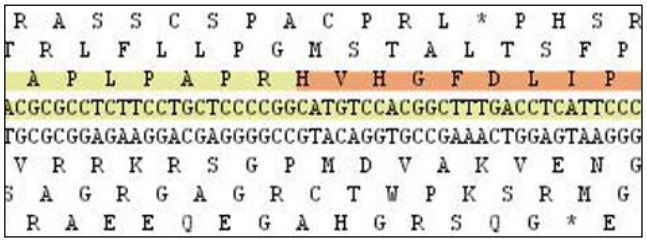

Figure 4: Resolution and detail for viewing sequences. The two nucleotides (DNA) sequences are surrounded by their associated amino acid (frame) sequences. A gene segment is highlighted in yellow to indicate its DNA sequence; the associated ORF segment is highlighted in yellow along its frame showing the amino acid sequence. Identified peptides are highlighted in red; the peptide sequences match their underlying ORF sequences.

full access to multiple resolutions at the same time as well as multiple views at the same resolution [17]. Window management is the major drawback of multiple views; linking the views and providing user-control of placement and size simplify window management. Suitable scales for each of the key resolution views are calculated from the input sequence length and window size.

The initial design of the bird's eye, or DNA, view was a square of 256 x 256 pixels that would depict an entire, albeit compressed, chromosome or plasmid. The compressed view has to encode as many pixels as possible while keeping the DNA pattern discernable. Since the direction of the DNA strands is important, as pointed out in Section 1.1, strand direction must be presented predictably. Our solution was to encode 128 rows of 256 pixels with the DNA, ORF and peptide information alternated with (thus, separated by) 127 rows of un-encoded (background) pixels. Rows are preferred over columns because horizontal lines are used in the traditional genomic graphics and because text is commonly read horizontally from left to right, but the information could be laid out similarly in columns. This layout provides 32K pixels to display an entire DNA unit. Using 130.4M as the average number of basepairs (bps) per human chromosome and 5M as the average number of bps per bacteria chromosome the "average" resolutions are 4K and 156 bps/pixel, respectively. If any part of a peptide or ORF falls into the sequence range of a given pixel, the pixel's properties reflect that information. This means that the pixels in this compressed view depict only the approximate location and size of peptides and ORFs. This slightly exaggerates their size. In practice, we found the 256x256 pixel square too small to read comfortably, so we doubled the scale. This seems to be the best compromise for a compressed view, see Figure 5, of an entire DNA unit suitable for navigation and context. This compressed DNA view provides distinct and important information in its own right. We call these views the DNA View (Figure 5), the ORF View (Figure 3), and the Sequence View (Figure 4).

5.2 Wrapped Line Metaphor

Our work on the DNA view led us to recognize the suitability of the wrapped lines as a metaphor for the DNA strands across all the views. At each key resolution, we have employed this metaphor to inject richer context and information content. The higher resolution views are not constrained by space to depicting the DNA as a single line but can provide additional information by including both strands and their frames. The related and aligned strands and frames as shown in Figures 2 and 3 are defined as a "tier."

In visualization, metaphors are used to model data by extracting the essence of the data and organizing it in a readily understandable model. This makes it easier for users to analyze and discover information hidden in massive data. However, there are risks with using metaphors. Metaphors should not over-simplify, over-complicate, or mislead; they should be suited to the task and data [18, 19]. Avise discusses metaphors from the perspective of a geneticist. He says that metaphors are valuable tools for thinking; they influence how we think about things; and we should evolve our metaphors or look for new ones as understanding of our problem space changes.

We have discussed how we see the wrapped line metaphor as a natural extension of the traditional genomic graphics as well as of the notion of genetic sequences as very long strings of text letters. We recognize several problems with the metaphor. First wrapping

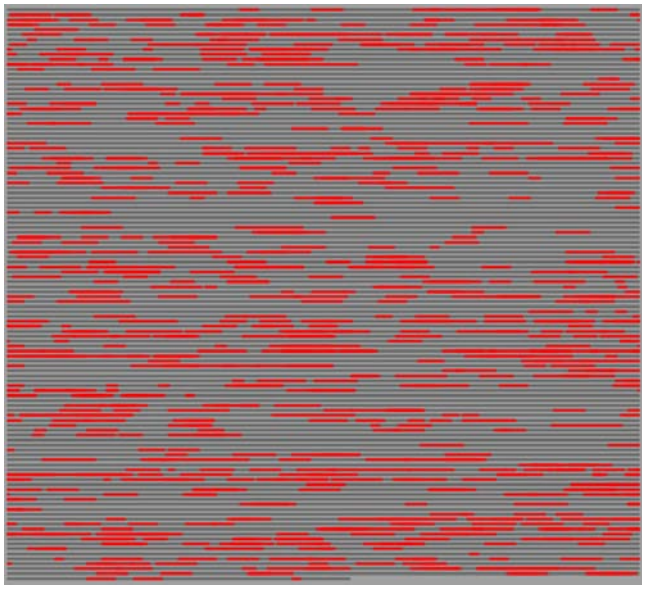

Figure 5: DNA view of an entire plasmid. This image shows only the location of the peptides relative to the DNA. Figure 9 shows the same

requires we interrupt the sequences at the right side of a view and resume them on the left. Thus, consecutive amino acids or nucleotides can be separated by the width of the view; intuitively continuous things ought to be connected. ORFs or peptides depicted by bars or lines may be similarity interrupted; very long ORFs could wrap twice or more. An alternative would be to maintain the continuity by making a U-turn at the sides of a view, then continuing in the opposite direction. In this way, the sequence could snake its way down the view in one contiguous line. This has the unacceptable disadvantage of confusing our sense of the DNA direction which would change from line to line.

In both layouts described so far, a second order discontinuity exists that further reduces their intuitiveness. Generally, the proximity of points in a display is assumed to indicate proximity of the underlying data. Note that neighboring pixels in the vertical direction represent sections of the sequence that are less related than more distant pixels along the same line. The Hilbert curve, or some modification, could be a suitable option for drawing a continuous line while minimizing the distance between neighboring pixels [20]. Again, this metaphor does not support the need for a sense of direction; advancing down the sequence could, at times, mean moving up, down, left, or right.

The failure to locate pixels representing neighboring DNA sections close together affects the selection of an area by rubber band. Consider a rubber band drawn as a small box left to right across 10 pixels per row for two rows in the center of the our DNA view. As the band is pulled from its starting line to the next, the selected area would include the upper leftmost pixel in the box to the end of the first line and the start of the second line to the lower rightmost pixel

of the box. The selected area would contain not 20 pixels, but 266 (256 + 10) pixels. While this is not intuitive, selecting just the 20 pixels inside the defined box would not make much sense; the result would be two loosely related segments of the sequence. The same problems with selecting an area by rubber apply with the U-turn and Hilbert lines.

One final problem with the wrapped line metaphor is that biologists are accustomed to seeing a bacteria chromosome depicted in a ring because this class of chromosomes is in fact circular rather than linear. Circular chromosomes are often drawn as a ring with the ORFs as radian segments across the ring. In the ring metaphor, the ORF sequences are discontinuous; the direction of the ORFs is hidden. The biggest disadvantage to the ring metaphor is the very high compression of the strand. Only a small portion of the available pixel space is used to represent the information. We believe that users will find the wrapped line more useful. From experience we know that providing a link to a familiar metaphor helps some users to trust and adopt new metaphors. We plan to add the ring metaphor at some time in the future.

In PQuad the default peptide and protein color encoding is the same for all resolutions, yellow for proteins and red for peptides. Figure 4 shows a full DNA unit wrapped in the compressed view with only the peptides shown. Figures 2 and 3 show an ORF and sequence tier, respectively. These tiers are wrapped for the full scale ORF and sequence views. The tiers are separated by space and, in the case of the ORF view, the tiers are visually distinguished by alternating the background colors. Examples of wrapped ORF views appear in the following sections. It is quite easy in at all three resolutions to distinguish ORFs with no peptides (all yellow), ORFs with a few peptides (yellow with some red), and ORFs almost completely covered by evidentiary peptides (mostly red).

5.3 Comparing Peptide Data Sets (Difference Visualization)

Differential proteomics is an important proteomic research area where peptide sets obtained from two or more different experimental conditions are compared. As discussed in section 4.3, there are a number of possible approaches for difference visualization. But based on users' needs, the preferred approach is to represent both peptide sets in the same views using color encoding to show the different cases. This has the disadvantage of limiting the number of sets we can compare at one time to two or, at most, three. But this is an acceptable limitation in view of the current manual spread-sheet methods. Color-coding the peptides and their parent ORFs distinguishes three cases: present in one condition, present in the other, and present in both. In the case of the ORFs, we must add a fourth case, no peptides.

Two colors are used to represent peptides produced by each of the two experimental conditions and a third color is used to represent the peptides produced by both conditions. Figure 6 shows the legend for difference views. This color scheme is consistent across the DNA and the ORF views. The typical sparseness of peptides and

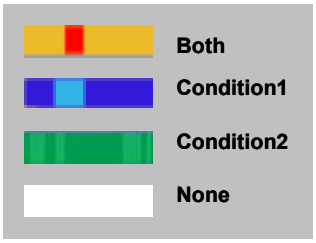

Figure 6: Legend for the difference visualization.

ORFs in the sequence view reduces the effectiveness of such coloring at that resolution.

With just a cursory glance at the DNA view of peptides only shown in Figure 7, we can easily distinguish the peptides produced by one (green), the other (blue), or both (red) conditions. An area of interest selected in the DNA view updates the ORF view to show the same area at the higher resolution with more detail as in Figure 8.

At first only the peptides were color encoded until it became clear that the collective conditions of the peptides associated with an

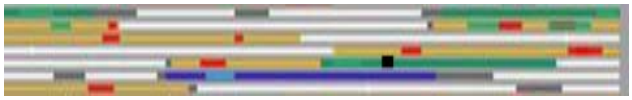

Figure 7: DNA view of difference visualization for peptides for Condition 1, Condition 2, and both conditions. Peptides are colored green, blue, and red, respectively. The black box shows the currently selected location.

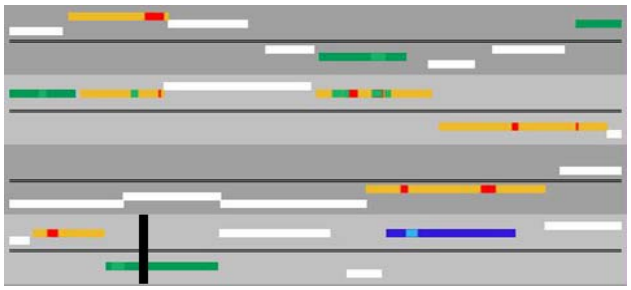

Figure 8: ORF view of the region surrounding ORF selected in the DNA view of Figure 6. The black box shows the currently selected location.

ORF ought to be propagated up to the ORF color encoding. Consider an ORF with two peptides, one from Condition 1 and the other from Condition 2. While the peptides are colored to indicate their individual conditions, the ORF is colored to indicate evidence from both conditions. Biologists will most likely be interested in studying the ORFs with peptides specific to a limited set of conditions thereby revealing information that might be used to identify or confirm protein function.

5.4 Filters and Queries

Both filters and queries are necessary capabilities in any exploratory analysis visualization. Users must be able to tune the visualization to suit their analytical task, shape the visualization to consider multiple perspectives, and control the amount of information presented. Figure 9 is the same as Figure 5 except the ORFs are visible. The ORFs are depicted as yellow lines overlaid by the red peptides. This reveals the ORFs' distribution and their relation to the peptides. The user can also filter out the peptides to view only the ORFs or filter out both ORFs and peptides to view the DNA alone. The user might choose any or all three options during an analysis to reveal alternate information or de-clutter the view.

Filters and queries can apply to ORFs and peptides. For example, an ORF-based query might request a display of the predicted protein function. Figure 10 shows the result of a request to see the TIGR-defined [16] protein functions; the ORFs are color encoded to indicate function. A combined color legend and histogram, shown in Figure 11 maps the colors to the function names while showing the relative distribution of ORFs across the function categories. A user can select function categories in the legend to highlight the ORFs with the selected function(s) in the view. There are public databases such as TIGR [9], GenBank [14], Kyoto Encyclopedia of Genes and Genomes (KEGG) [21], and Gene Ontology (GO) [22] that contain

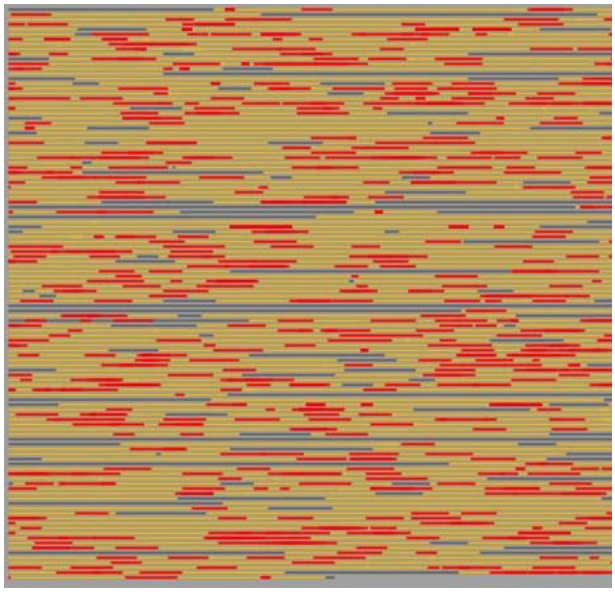

Figure 9: DNA view of peptides and ORFs. This is the same as Figure 4 except the ORFs are now visible.

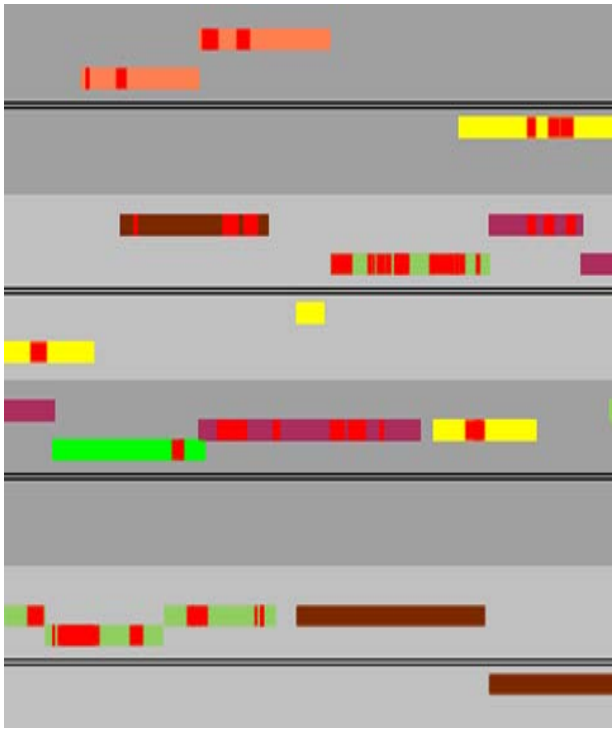

Figure 10: ORF view with ORFs color-coded with TIGR-defined function. The combined legend histogram maps the colors to functional categories.

categorical information about ORFs that can easily be depicted by color. Using color to encode categories limits the number of categories to between 7 and 9, the number of colors that can be distinguished by the human eye at a glance[23]. For this reason, we are investigating alternate ways to encode categories, for example, combining texture and color.

A peptide-based query might be a request to color-encode peptides in the visualization based on peptide identification confidence

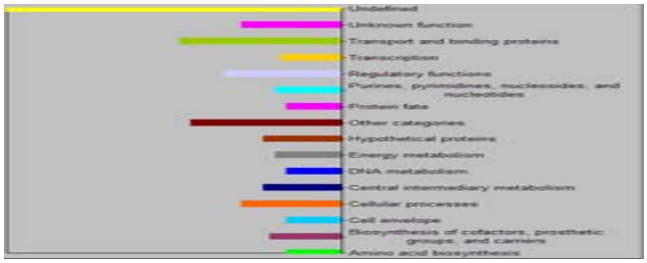

Figure 11: ORF function legend/histogram.

metrics. Since the "identifications" are actually predictions and the predictions are made with varying levels of confidence, showing the confidence metrics is important for some analysis tasks. Biologists might use the peptide prediction metrics to distinguish the highly likely from highly unlikely peptide evidence, to evaluate peptide identification software, or to test algorithms for confirming the presence of proteins based on peptide evidence.

5.5 Modelling the Data

The size of the data for the PQuad visualization can be huge. We decided to generate the amino acid sequences from the DNA sequence on the fly as needed rather than store or look up the sequences for all the proteins and peptides. Given the primary strand, the complement strand is easily generated. Given the bounding indices of an ORF or peptide relative to its strand, generation of the relevant frame sequence is straightforward. This approach has many advantages with only a few disadvantages for the bacterial data we have used so far.

For the nominal data set, we need the nucleotide sequence of the primary strand of a chromosome or plasmid and the ORFs and the peptides derived from that sequence. Rather than store all the ORF and peptide sequences (a truly huge amount of data), these sequences are reduced during the initial ingest to bounding indices relative to the nucleotide sequence. The ORF and peptide sequences are then generated on demand from the nucleotide sequence and bounding indices. The first data set was based on the chromosome of *Shewanella oneidensis* (*ShewO*). This chromosome has 4,968,865 nucleotides, 4781 TIGR-defined ORFs, and 738 identified peptides.

One problem with the initial implementation using this approach is that during the transcription, some nucleotides may be skipped. This is called a *frame shift* since the protein sequence changes from one frame to another. In this case, parts of our generated amino acid sequences will be incorrect. The generated sequence for an ORF with a frame shift will match the actual ORF sequence only to the location of the skipped nucleotide; the remainder of the generated sequence is incorrect. For the *ShewO* chromosome, only one such ORF exists out of the 4781 ORFs. The solution is to define multiple bounding index pairs to describe ORFs with skipped nucleotides as a series of segments. The big challenge is in depicting these ORFs.

Reducing all the peptide and ORF sequences to bounding indices in the nucleotide sequence allows us to operate globally on a single, simple reference scheme. All links between the DNA, peptide, and ORF sequences are through the nucleotide sequence indices. To draw a sequence section or report information about the area under the current cursor position, we query the peptide and ORF collection classes using an index pair to retrieve lists of the relevant ORFs and peptides. To do this, we create (once) two hash maps using binned nucleotide indices as the key and a hash set of ORF/peptide indices associated with the nucleotide index as the value object. To retrieve the list of ORFs from the ORF hash map, we submit an index pair that defines a sequence interval and receive an iterator over the set of candidate ORFs. The candidate ORFs must then be queried to see if they are indeed inside the target interval.

The number of nucleotides is much too great to map from each nucleotide index to the set of peptides or ORFs associated with it. Even if the number of sequence indices were not too large, there would be entirely too much duplication; for instance, if an ORF falls between indices 7 and 247, we would need to save the index of that ORF for each of the 248 nucleotide indices. So we bin the indices and use the bin number as the key. There needs to be a balance between the number of bins and the size of the hash sets. This could use more investigation; for the data we are using, 9000 bins seem to work well.

5.6 Providing Contextual Information

PQuad provides contextual information at multiple levels including not only graphical context but also descriptive information about the data currently viewed; derived information such as counts of ORFs and peptides, the length of the DNA sequence, and the current view resolution; legends; selection location indicators across views; and visual query (querying by brushing, that is, moving the cursor over, an area in the visualization). Figure 12 shows the descriptive information panel.

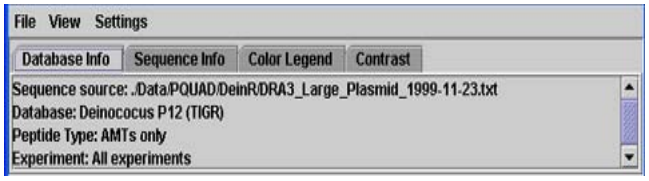

Figure 12: Descriptive Information Panel. There are two tabbed panes for the quantitative and qualitative information about the current data set.

For browsing, our users generally want more information than will easily fit in a small label. Drawing sizeable labels near the cursor would occlude too much of the graphic. For this reason, PQuad displays information related to the current cursor position in a separate panel below the graphic. PQuad's Visual Query panel, shown in Figure 13, provides the DNA sequence index range of the pixel under the cursor, the list of collocated ORFs and peptides in the first, second, and third lines, respectively. For now, the information choices are fixed. In the future, users will be able to specify the information presented. For instance, biologists may prefer to see the peptide sequence rather than peptide name or the protein function label rather than the frame number.

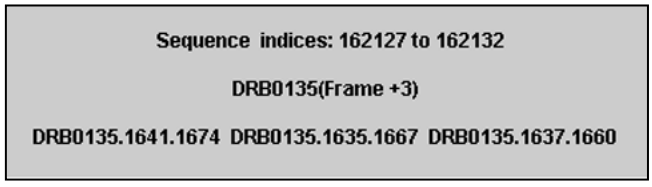

Figure 13: The Visual Query panel located at the bottom of all three views.

6 Formative Evaluation

After implementing the initial prototype of PQuad with the three views at different levels of resolution and detail, we met with systems biologists for informal formative evaluations. The response was encouraging. All indicated an interest in using PQuad in their research. For one, we implemented the capability to export the Sequence view of an entire chromosome to multiple files. The biologists had a number of suggestions for added functionality. It is clear that they have a variety of research data and interests as well as personal preferences. As might be expected, some of their views are conflicting. Below we discuss two issues raised during the evaluations.

First, one biologist wanted a highly compressed window that would show the current position of the active cursor in all open views. At the time, only selections were communicated between views. Brushing information was local to a view. We can add the capability to instantly update all views to show current cursor movement. While this might work nicely when browsing a higher resolution view, it would be chaotic when browsing over a lower resolution view. A small change in cursor position in the DNA view, for example, would force the ORF and sequence views to continuous refocus and redraw. We have implemented a browser that shows the brushed cursor position in the DNA View in a small window at the ORF view resolution. As the user brushes across a DNA view, for example Figure 4, not only does the visual query area provide current sequence indices and peptide and protein names, the browser shows a continuously updating view at the next higher resolution and level of detail, for example Figure 3.

Second, another biologist wanted to see protein function information depicted by coloring the ORFs. For a subsequent version of our prototype, we downloaded this information from TIGR and colored the ORFs based on their functions, as discussed in Section 5.4. The result is seen in Figure 10. Upon showing screenshots to other biologists, they advised us that ORF function information was useless to them; they would prefer cellular location or pathway information from KEGG. This illustrates that the needs and preferences of the biologists differ widely. Our requirements to appeal to biologists and to supply appropriate context imply a generalized ability to filter on whatever data the biologists can supply.

7 Summary and Future Work

PQuad provides powerful analysis capabilities through the novel visualization of high-throughput proteomic data. We have defined the three key resolutions for viewing peptides identified from MS experimental data. PQuad provides these resolutions through coordinated multiple views. It employs a wrapped line metaphor to DNA sequences across all views to provide a larger context for exploring and analyzing the peptide data. In addition, PQuad supports *differential proteomics* by simplifying comparison of peptide sets from different experimental conditions.

Scalability is a major challenge. At this time PQuad easily handles DNA of 5 million basepairs with up to several thousand proteins and peptides. PQuad bogs down as the DNA or proteomic data sets increase in size both in the time to preprocess and in the refresh rates of the visualization. We continue to seek ways to optimize PQuad. Improved indexing is one solution. The human genome is much more complex than the bacteria genomes currently visualized. Unlike bacteria genes, human genes have large areas that are not translated; the wrapped line metaphor may not be suitable for the human genome. It will be interesting to continue studying the wrapped line metaphor in this context to better understand its strengths and limitations. Presently, PQuad ignores the frame shift problem mentioned in Section 5.5. Even though the biologists seem unconcerned about this, we plan to implement multiple bounding index pairs as discussed. The level of customization, in terms of auxiliary data, needed by biologists presents an interesting challenge. PQuad needs user-friendly and dynamic solutions to problems associated with importing diverse, related data files and integrating this data into the effective visualizations. Finally, there a number of issues that should be more formally tested including the multiple view approach and PQuad's use of colors.

PQuad is on the road to adoption by biologists. Several early adopters are interested in analyzing their proteomic data with PQuad. The input formats for PQuad are simple, enabling users to

create input files from their customary spreadsheets; a new system under development [24] will soon be delivering data from the Pacific Northwest National Laboratory peptide database to researchers in PQuad-ready format.

ACKNOWLEDGMENTS

This work was supported by the U. S. Department of Energy through the Computational Sciences and Engineering Laboratory Directed Research and Development program at Pacific Northwest National Laboratory (PNNL). PNNL is a multi program national laboratory operated by Battelle for the U.S. Department of Energy under Contract DE-AC06-76RL01830.

REFERENCES

[1] D. W. Speicher, "Proteomics: an infinite problem with infinite potential," *The Scientist*, vol. 16, pp. 12, 2002.

[2] J. R. Yates, J. K. Eng, A. L. McCormack, and D. Schieltz, "Method to correlate tandem mass spectra of modified peptides to amino acid sequences in the protein database," *Analytical Chemistry*, vol. 67, pp. 1426-1436, 1995.

[3] M. Mann and M. Wilm, "Error-tolerant identification of peptides in sequence databases by peptide sequence tags," *Analytical Chemistry*, vol. 66, pp. 4390-4399, 1994.

[4] J. K. Eng, A. L. McCormack, and J. R. I. Yates, "An approach to correlate tandem mass spectral data of peptides with amino acid sequences in a protein database," *Journal of the American Society for Mass Spectrometry*, vol. 5, pp. 876-989, 1994.

[5] S. K. Card, J. D. Mackinlay, and B. Shneiderman, *Readings in Information Visualization: Using Vision to Think*. San Francisco: Morgan Kaufmann Publishers, Inc, 1999.

[6] GeneSpring, "Silicon Genetics," vol. http://www.silicongenetics.com/cgi/SiG.cgi/Products/GeneSpring/index.smf. Redwood City, CA.

[7] OmniViz, vol. http://www.omniviz.com/. Maynard, MA.

[8] Spotfire, Somerville, MA.

[9] TIGR, "The Institute for Genomic Research," http://www.tigr.org/.

[10] L. S. Heath and N. Ramakrishnan, "The Emerging Landscape of Bioinformatics Software Systems," *IEEE Computer*, pp. 41 - 45, 2002.

[11] J. D. Jaffe, H. C. Berg, and G. M. Church, "Proteogenomic mapping as a complementary method to perform genome annotation," *Proteomics*, vol. 2004, pp. 59-77, 2004.

[12] K. Rutherford, J. Parkhill, J. Crook, T. Horsnell, P. Rice, M. A. Rajandream, and B. Barrell, "Artemis: sequence visualization and annotation," *Bioinformatics*, vol. 16, pp. 944-5, 2000.

[13] EMBL-EBI, "European Bioinformatics Institute," *http://www.ebi.ac.uk/embl/*.

[14] GenBank, "GenBank: NIH genetic sequence database," *www.ncbi.nlm.nih.gov/*.

[15] T.-M. Rhyne, J. P. Lee, D. Carr, G. Grinstein, J. Kinney, and J. Saffer, "Visualization Viewpoints: The Next Frontier for Bio- and Cheminformatics Visualization," *IEEE Computer Graphics and Applications*, vol. 2002, pp. 6 - 11, 2002.

[16] D. F. Jerding and J. T. Stasko, "The Information Mural: A Technique for Displaying and Navigating Large Information Spaces," In *Proceedings of IEEE Symposium on Information Visualization 1995*, Atlanta, Georgia, 1995.

[17] C. Plaisant, D. Carr, and B. Shneiderman, "Image-Browser Taxonomy and Guidelines for Designers," *IEEE Software*, vol. 12, pp. 21-32, 1995.

[18] T.-M. Rhyne, "Evolving Visual Metaphors and Dynamic Tools for Bioinformatics Visualization," In *Proceedings of IEEE Visualization Conference 2002*, Boston, Massachusetts, 2002.

[19] J. C. Avise, "Evolving Genomic Metaphors: A New Look at the Language of DNA," *Science*, vol. 294, pp. 86-87, 2001.

[20] P. C. Wong, K. K. Wong, H. Foote, and J. J. Thomas, "Global visualization and alignments of whole bacterial genomes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, pp. 361- 377, 2003.

[21] KEGG, "Kyoto Encyclopedia of Genes and Genomes."

[22] GO, "Gene Ontology."

[23] C. G. Healey and J. T. Enns, "Large datasets at a Glance: Combining Textures and Colors in Scientific Visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, pp. 145-167, 1999.

[24] S. Havre, M. Singhal, B. Gopalan, D. Payne, K. Klicker, G. Kiebel, K. Auberry, E. Stephan, B.-J. Webb-Robertson, and D. Gracio, "Integrating Evolving Tools for Proteomics Research," In *Proceedings of International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences*, Las Vegas, NV, 2004.