A Visual Analytics Approach to Exploring Protein Flexibility Subspaces

Scott Barlowe
UNC-Charlotte

Jing Yang*
UNC-Charlotte

Donald J. Jacobs
UNC-Charlotte

Dennis R. Livesay
UNC-Charlotte

Jamal Alsakran
Kent State University

Ye Zhao
Kent State University

Deeptak Verma
UNC-Charlotte

James Mottonen
UNC-Charlotte

ABSTRACT

Understanding what causes proteins to change shape and how the resulting shape influences function will expedite the design of more narrowly focused drugs and therapies. Shape alterations are often the result of flexibility changes in a set of localized neighborhoods that may or may not act in concert. Computational models have been developed to predict flexibility changes under varying empirical parameters. In this paper, we tackle a significant challenge facing scientists when analyzing outputs of a computational model, namely how to identify, examine, compare, and group interesting neighborhoods of proteins under different parameter sets. This is a difficult task since comparisons over protein subunits that comprise diverse neighborhoods are often too complex to characterize with a simple metric and too numerous to analyze manually. Here, we present a series of novel visual analytics approaches toward addressing this task. User scenarios illustrate the utility of these approaches and feedback from domain experts confirms their effectiveness.

Index Terms: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques; J.3 [Computer Applications]: Life and Medical Sciences—Biology and genetics

1 INTRODUCTION

Deciphering how a protein's function changes as its three-dimensional shape is altered provides insight for making improved drugs and protein related therapies. A major factor influencing alterations in shape is the ability of protein subunits, or *residues*, to change shape. Scientists can characterize this ability, referred to here as *protein flexibility*, with a variety of computational models that are based on the mechanical properties of residues under different parameterized conditions. The challenges encountered when trying to explore the many model outputs that represent the mechanical variations influencing flexibility can be slow and may deter understanding. Difficulties include exploring the possible states for each parameterized condition and then linking their influence on local collections of residues, or *protein subspaces*, to global behavior.

To illustrate the challenge of protein subspace exploration, we present the following scenario of exploring allosteric response, namely how local protein regions behave in concert or in isolation to affect change across a protein. The study of allostery allows scientists to uncover the intricate and hidden relationships among protein subunits that are critical in altering a protein so that it exhibits desired behavior. In this scenario a computational model is used to simulate how residue flexibility changes as each residue is perturbed and the changes are visually presented in plots (see Figure 1 for an example). Plots like these are common in other parts of bioinformatics, such as in molecular dynamics simulations [19], and present similar challenges:

Identifying subspaces of interest. A scientist begins with studying how parameters influence changes in residue flexibility. A good starting point is to simultaneously isolate and explore residue flexibility in a protein subspace and then compare the effect of different parameters. In the model outputs, such a subspace occupies the same location in different plots, each of which represents the flexibility for a single parameter combination. Subspaces may vary from a large sequence of residues to individual residues. Through exploration, she wishes to identify a few subspaces that represent interesting behavior, such as those experiencing the greatest or least change in flexibility when parameters change. In particular, she is interested in the two subspaces that bound the opposite ends of the flexibility range.

Examining subspaces within neighborhoods. After identifying several subspaces of interest, the analyst includes neighboring residues in the analysis. Examining subspaces within their neighborhoods allows her to assess if a parameter's effect on a single region extends to neighboring residues and to more precisely define the range of the region with interesting behavior. Additionally, it allows her to learn how the residues of interest are influenced by and act in concert with other residues.

Grouping subspaces. The analyst is now ready to examine residue similarity as parameter values are varied. Grouping similar residues and noting any changes in their similarity is useful for precisely locating where parameters have different effects on individual residues. For example, suppose that there are several residues thought to be similar and group together for most parameter combinations. If a parameter combination is input to the model and the similarity changes noticeably for one or more residues, the analyst can isolate the residues with decreased similarity. Then, the isolated residues can be studied more closely under the latter parameter set to understand the differences in flexibility. Additionally, this information can be used to test the correctness of the model against domain knowledge.

Similar scenarios can be found in numerous other applications, such as in the study of the effect of mutations or substrate binding, as well as using covariance plots from molecular dynamics simulations. These tasks are difficult with current tools. A single protein can often have hundreds of residues and the flexibility of each residue may vary with each possible parameter combination. The size of subspaces and the influence of neighboring regions can also vary. Computational methods applied during analysis often result in summary metrics that hide underlying relationships. Parameter optimization tools are ineffective because neither the optimal conditions nor the parameter values that result in those conditions are known beforehand. Visualization tools are limited by the enforcement of a strict ordering along domain plot axes and the lack of functions for including neighborhoods into detailed analysis.

In this paper we present a novel approach to help scientists more efficiently and effectively investigate protein subspaces. Our approach is built upon an existing system that aims to ease discovery in protein flexibility outputs, a framework called WaveMap

*Corresponding author: jyang13@uncc.edu

IEEE Pacific Visualization Symposium 2013
26 February - 1 March, Sydney, NSW, Australia
978-1-4673-4797-6/13/$31.00 ©2013 IEEE

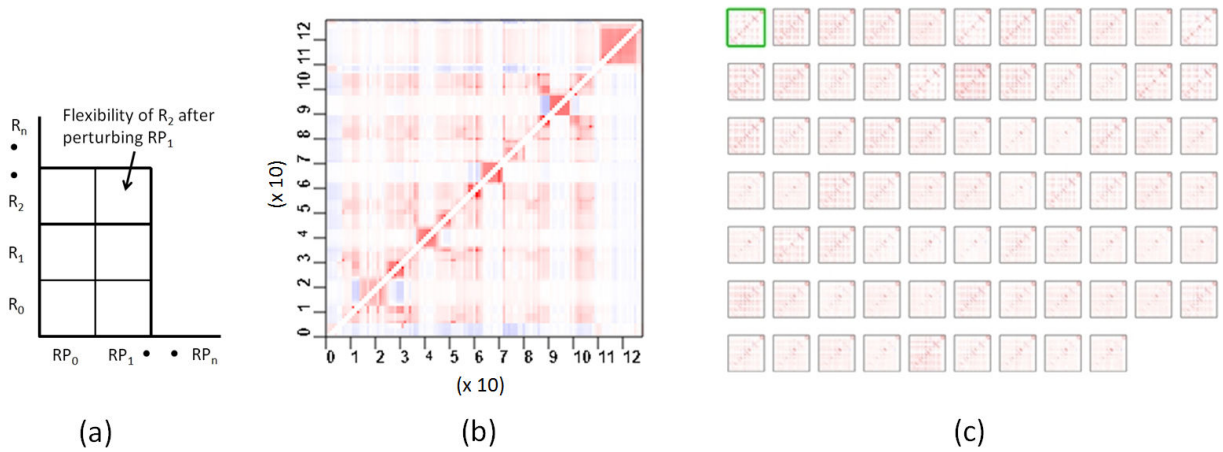

Figure 1: The layout of a flexibility plot and the CheY data set as shown in [4]. (a) Flexibility plot layout where each axis is ordered by the residue sequence occuring in the three-dimensional protein space. At each index is the flexibility measure of residue *R* after *RP* is perturbed. (b) A color-coded flexibility plot based on the layout in (a) and containing 128 rows and 128 columns. One plot encodes the response resulting from a single set of model parameters. In each column, blue indicates rigid areas and red indicates flexible areas. (c) The entire data set consists of 75 plots, each one representing a single parameter combination. The subtle and faint patterns are difficult for analysts to investigate.

[4]. The original framework lacked the capabilities needed to conduct practical visual exploration tasks in protein subspaces. Plot-carving, sliding subspaces, and tailed transitions are new functions which allow the completion of the above high-level tasks which was previously almost impossible.

Plot-carving allows scientists to explore subspaces and individual residues and identify interesting ones according to their behaviors in varying parameter sets. Analysts can interactively split the data into subspaces and individual residues. Histograms showing the distribution of flexible and rigid residues within a local area then help the analysts categorize collections of residues influencing overall subspace flexibility.

Sliding subspaces facilitate the tracing of flexibility changes in relation to a subspace's neighborhood. In this view, users can interactively scan subspaces adjacent to a focus subspace to observe how the similarities among different plots change within these subspaces. This presents domain analysts with a more complete representation of a subspace as it relates to its surrounding flexibility environment. In addition, snapshots are provided so that users can examine the trends in a static view.

Tailed transitions utilizes a force-directed layout to help scientists group individual residues by the similarity of their flexibility and examine how the grouping of residues changes as parameters are varied. Tails are attached to the nodes representing single residue behavior, highlighting the change from one parameter set to another.

The paper is organized as follows. Background on the computational model, flexibility plots, and an early approach to their visual analytics are presented. Related work in subspaces is then discussed and followed by the high-level tasks of domain analysts for subspace exploration. Next, our new functions are described in detail. Along with each description, example scenarios are presented illustrating system use. Finally, we report feedback from domain experts and list future work. (Several parts of this paper are derived from one of the authors' dissertation [3]. Readers interested in further details are referred to that document.)

2 Background

2.1 Flexibility Plots

Flexibility plots are widely used to convey the flexibility measures of proteins. For example, Figure 1 shows a set of flexibility plots representing the output of the Distance Constraint Model (DCM) [15], [13], a computational model used to calculate a protein's ability to change shape based on the relationship between energy and mechanical constraints. In this case, the plots convey the allosteric response of the CheY protein under a variety of parameter sets, one plot for each parameter set. In each plot, the structural information is encoded along each axis by the amino acid (residue) sequence. In each column the residue corresponding to the current location in the residue sequence is altered, or *perturbed*, and the flexibility of each remaining residue in the sequence is plotted along the column (Figure 1(a)). Parameters include variations in the energy of a hydrogen bond between a protein and a solvent, an indicator of the torsion interactions in the native state of local regions, and a measure of torsion angle entropy. Each plot represents the allosteric response across a protein for one set of parameter values. In the plot shown in Figure 1(b), blue indicates rigid locations and red indicates flexible locations. Understanding the intricate relationships among individual residues and local groups of residues in a single plot is complex. Extending the analysis to an entire data set (Figure 1(c)) is almost impossible with current tools.

Barlowe et al. [4] developed a system that was among the first visual analytics approaches allowing users to effectively explore a large number of flexibility plots simultaneously. WaveMap was primarily tested on allosteric data resulting from the DCM [15], [13] and used wavelet analysis within a highly interactive framework to reduce the data to localized neighborhoods where flexibility behavior abruptly changed. Users were allowed an overview of the features across the data set, neighborhood analysis for a small number of plots, and a detailed contextual setting considering only one, fixed height column for each plot. WaveMap was evaluated by researchers working with the DCM, who requested that it be extended to include subspace exploration features since it is an important task that was not able to be addressed in the original version.

2.2 Subspaces in Visualization

There are multiple works addressing interactive analysis or construction of subspaces in large data sets. Several of those works are now presented. Ferdosi and Roerdink [9] present approaches based on subspace clustering and ranking to help alleviate the shortcomings caused by the ordering constraint in parallel coordinates and the clutter found in scatterplot matrices. Low quality subspaces are iteratively culled and user interactions include the ability to change the ordering of the scatterplots by dragging and dropping dimen-

sions.

Guo et al. [11] present a platform based on interactive feature selection. Automatic techniques along with a color-coded entropy plot helps users find interesting subspaces. Another plaform allows users to isolate subsets of data according to linear trend discovery and to choose which variables are independent and which are dependent [12]. VISA (Visual Subspace Clustering Analysis) [2] attempts to occlude redundant spaces with efficient representations and allows subspaces to be browsed according to a normalized distance function. Subspace matrices can be viewed where rows represent clusters and columns represent dimensions. The major limitations of the above works and many other subspace approaches for the application here are listed below:

Reliance on reordering. Reordering subspace components in flexibility plots will disrupt the sequence information encoded by each axis. Individual residue behavior does not occur in isolation and may have large influence on the behavior of other (adjacent or non-adjacent) residues.

Exploration of subspaces and their context. Ferdosi et al. [8] note that too many visualization approaches only aid in presenting cluster analysis and do not help in exploring individual subspaces. In addition, the contextual information of subspaces under examination is omitted in most existing approaches. Because of the contextual significance of the residues on the three-dimensional behavior of proteins, the inability to explore neighboring regions on a residue-level basis is a critical limitation.

We provide a unique approach to subspace exploration when there is an order constraint and the contextual information of subspaces is a focus of the analysis. Subspace displays and similarity functions maintain the spatial significance of neighborhoods. Some views allow the user to order subspaces according to similarity but only in support of the primary operations that preserve the structural order encoded by plot axes. We utilize covariance and bin-by-bin histogram similarity measures which are intuitive and simple to implement. Finally, our approach provides access to individual residue flexibility values and individual data histogram bins across the entire data set.

2.3 Histogram Similarity

Histograms have been successfully integrated into visual analytics systems. For example, Guo et al. [12] used histograms to display distributions of distances for finding linear trends in model variables. Barlowe et al. [5] used histograms to visually explore partial derivatives so that highly correlated variables can be detected and used in interactive model building. Our approach integrates histograms into a visual analytics approach to help categorize subspaces in protein flexibility plots. Furthermore, we propose a set of interactions for histograms, such as a trim tool that allows the elimination of bins for more targeted comparison.

In our approach, histogram similarity is calculated for comparing histograms of multiple subspaces. There are many algorithms for calculating histogram similarity. Surveys of histogram bin measures can be found in [18] and [14]. Most types of histogram distance measures can be classified as either bin-by-bin, cross-bin, or a hybrid of these. The histogram similarity measure chosen for this work was χ^2, a bin-by-bin distance calculation shown below:

$$d_{\chi^2}(H,K) = \sum_i \frac{(h_i - m_i)^2}{m_i} \quad (1)$$

In Equation 1, d represents the distance between two histograms H and K, where h_i is the number of data items in the i^{th} bin of histogram H, and m_i is the average size of the i^{th} bins in H and K. The reasons for choosing a bin-by-bin distance are discussed later and will become more clear as the approach is described.

3 Requirement Analysis

Domain experts who provided the motivation for this work consist of two faculty researchers, a research scientist, and a graduate student. They have have been collaborating with visualization experts for over four years on the problems associated with flexibility plot outputs. Requirements were communicated and feedback gathered from domain scientists through email, face-to-face meetings, and video demonstrations. These are explained below:

Identifying subspaces of interest. Scientists need to compare the flexibility of subspaces that occupy corresponding locations across the data set as well as the flexiblity of individual residues within those subspaces. By providing an efficient method of comparison, scientists can find meaningful similarities (or dissimilarities) and unexpected abnormalities quickly. For the model outputs described in this work, boundaries of the subspace of interest need to be transposed onto the other plots so that the flexibility values and the flexibility distribution within an area occupying the same three-dimensional protein space can be analyzed simultaneously.

Examining subspaces within their neighborhoods. Parameter influence may vary according to subregion size, location, or parameter combination. Knowing how a given subspace behaves within the context of its neighborhood for a set of parameters will aid scientists in performing more efficient hypothesis testing and developing more accurate models. Currently, scientists employ an ad-hoc method of mentally trying to include adjacent rows and columns without any guidance as to what should be included and without any feedback regarding the impact of neighboring regions.

Grouping subspaces. Scientists should be able to systematically browse parameter variations and observe the change in residue similarity. Systematic browsing facilitates precise location and defining of parameter behavior. To accomplish this task, scientists are required to partition the plots into individual columns, calculate column similarity within a single plot, and then manually note the differences in similarities for different parameter combinations.

In the following sections, we present our approaches toward addressing the above tasks. The CheY dataset is used to illustrate our approaches, but they are general enough to be applied to other datasets for which the above subspace exploration tasks are desired. In the CheY dataset, there are 75 flexibility plots, representing the allosteric response of the CheY protein under 75 different parameter sets. They need to be analyzed simultaneously to understand how different parameter sets influence shape alterations.

4 Identifying Subspaces of Interest

4.1 Plot-carving

A visual analytics approach named *plot-carving* has been developed to allow users to define subspaces, examine subspaces or individual residues occupying corresponding locations for all parameter sets in either flexibility values or histograms, and thus identify subspaces and residues of interest for further analysis. This approach utilizes a combination of visualizations, intuitive interactions, and automatic analysis.

Users first interactively define the subspaces by carving the plots, namely to divide the plots into subregions (subspaces) using a grid, whose number of horizontal and vertical divisions are interactively controlled by the user. Figure 2(b) shows an example of a carved plot with the carving grids displayed.

Then, the users can click a grid cell on the carved plot to select a subspace for further analysis. They can choose a grid cell three ways. First, they may be interested in a certain subspace because of prior domain knowledge. Second, they may use the previously described wavelet analysis to guide them to areas of interest. Third, they may systematically click on adjacent grid cells until a subspace of interest is found.

After selection, corresponding subspaces in all the plots are displayed (see Figure 2(c)). We name each of them a **subspace plot**.

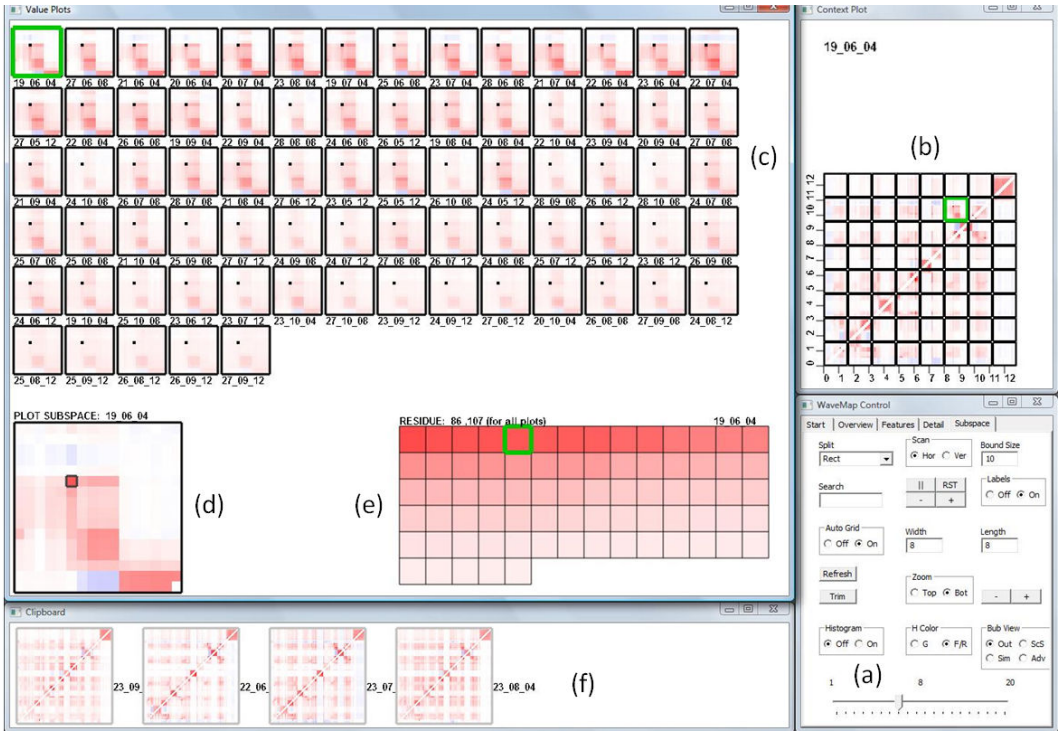

Figure 2: (a) Control panel with input boxes for changing the length and width of grid cells and the size of the bounding window. (b) The grid divides the data set into subspaces that can be selected by clicking. (c) Subspace plots occupying the same location across the data set as the selected subspace in (b). Subspaces may vary in both the location of flexible areas and their degree of flexibility. The selected subspace becomes the target plot and is highlighted. (d) Enlarged view of the selected subspace. An individual residue has been selected and highlighted. (e) The flexibility value of the current residue sorted across the data set. Clicking on a residue square changes the target plot. (f) A clipboard holds plots containing subspaces of interest.

They can be examined by viewing either the original residue flexibility values, as in Figure 2 (c), or histograms, which are further explained in Section 4.2.

Once a subspace of interest has been identified, individual residues can be investigated. A current residue is selected as the user clicks within an enlarged display of the current subspace plot being examined (Figure 2(d)). The current residue is outlined in black and is marked in all of the subspace plots by a black dot (Figure 2(c)). The flexibility values for the current residue across all the subspace plots are sorted in the far right portion of the main display with the current plot labeled and highlighted (Figure 2(e)). Middle-clicking a subspace plot sends its entire plot to the clipboard (Figure 2(f)) for further investigation in other views.

4.2 Histogram View

Histograms representing the local flexibility distribution can be displayed instead of the original values (Figure 3(a)) and utilized to categorize the residues within each subspace as flexible or rigid. Each bin of a histogram represents a flexibility range and its height represents the proportion of residues whose values fall into that range. Bin color is determined by the average flexibility of its members where bins to the left are more rigid and bins to the right are more flexible.

There are many interactions available for both entire histograms and individual bins. They are listed below:

Selection and Sorting. Clicking on a histogram will highlight it in a green box and make it the target. The remaining histograms will be sorted based on similarity to the target which eases comparison. The histograms will be arranged in rows where the target is in the top left corner of the screen and the histograms to the right of the target are progressively less similar. Middle-clicking a histogram will place the entire plot containing that subspace on the clipboard.

Dynamic Bin Sizes. Slider movement allows the number of bins in a subspace to be increased or decreased. Histogram similarity is recalculated after each change in bin size.

Bin Trimming. Trimming the histograms allows residues in selected bins to be removed from the data set so that only the subset of bins that are of interest can be compared (Figure 3(b) and 3(d)). The residue flexibility values in removed bins are filtered from the original data value subspaces (Figure 3(c)).

Histogram similarity measures utilize a bin-by-bin histogram measure (Equation 1). There were several reasons for this decision. First, bin-by-bin measures generally have a lower complexity than the other distance types [6]. Second, a method for determining similarity among bins having the same flexibility range is desired. A method for pair-wise comparison could have been chosen to preserve perceptual similarity [18]. However, perceptually similar histograms that are identical but differ only by a shift left or right would destroy any meaningful flexibility information encoded by bin color (Figure 3(e) and 3(f)). Third, the sensitivity to bin size often noted as a disadvantage is controlled by letting the user interactively choose the number of bins.

Example 1.1: An example now illustrates how some of the above functions are able to aid scientists in exploring subspace data sets. Mottonen et al. [15] found that allosteric response is both conserved and variable across the CheY protein. To investigate this further using the added subspace functions, each plot is divided into eight grid cells along both the width and length. In this case, the grid is browsed row by row and sections are investigated individually for their consistency. Many subspaces visited are consistent across the varying parameters with almost identical histograms.

Other subspaces seem to exhibit more variable behavior. The

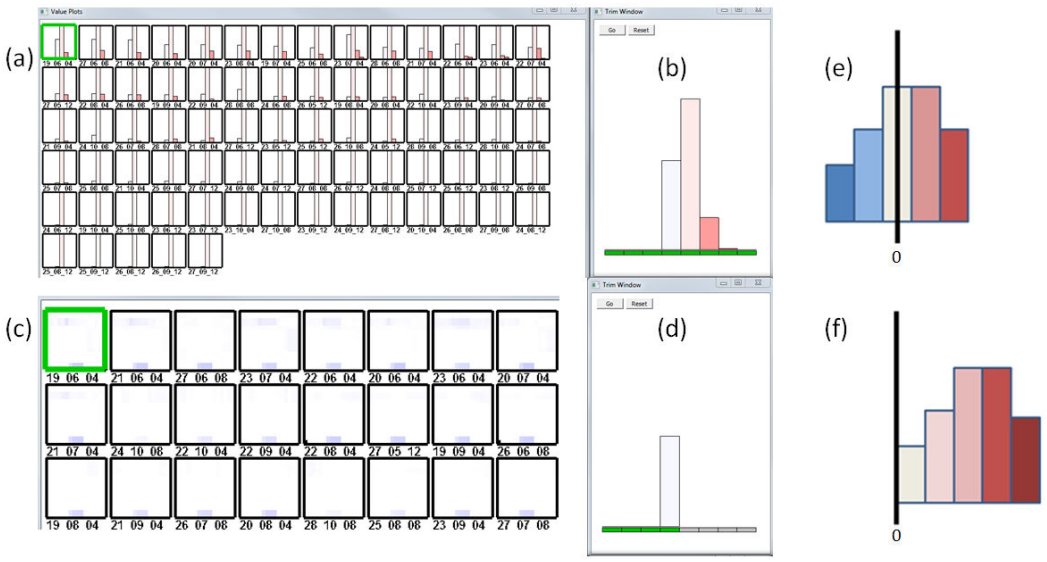

Figure 3: (a) Histogram view sorted by bin-by-bin distance. The number of bins is user-defined and colored according to average flexibility of its members. (b) The trim tool allows the elimination of bins for more targeted comparison. (c) The original data items after the red bins are eliminated by the trim tool. (d) Trim tool after bin removal. (e) - (f) Perceptual similarity between two histograms may not ensure that encoded flexibility meanings are preserved.

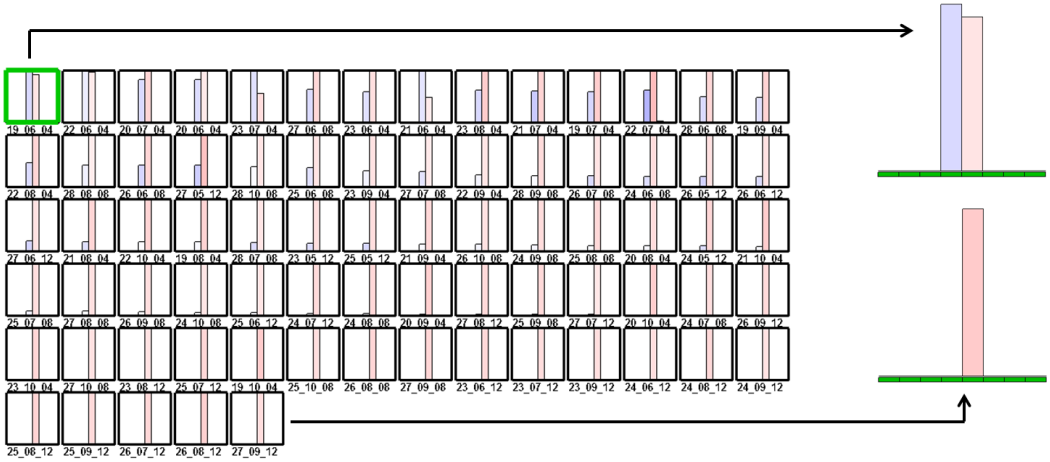

Figure 4: Histograms show a subspace where flexibility has a wider range (beginning of the sort at top left) and a narrow range (end of the sort).

histogram view confirms this observation for one such subspace. In Figure 4, the top left plot has a rigid bin having more members than the flexible bin. However, the rigid bin (blue) generally becomes smaller as the sequence of histograms progresses towards the end of the list. At the end of the list, there are no rigid residues. This variation in histograms for a single, corresponding subspace indicates differences in flexibility when changing parameter combinations.

The domain analysts wish to explore these differences in more detail. A subspace plot from the mixed (variable) group and another from the consistently flexible group (the two histograms on the right of Figure 4) are placed on the clipboard to be propagated to subsequent views. By investigating the extremities of the sorted list, the entire range of parameter effects can be observed in the sliding subspaces view (see Section 5). Adding more plots in the middle of the list to the clipboard will help reveal more subtle patterns.

5 Examining Subspaces within Neighborhoods

5.1 Sliding Subspaces

After a subspace of interest is identified, its neighborhood is examined through an interaction named *sliding subspaces*. In particular, right-clicking a subspace in the carved plot starts an automatic window sliding over the plot which defines a set of subspaces surrounding the one clicked (Figure 5(b)). For each subspace encountered during the slide, the similarities between the plots on the clipboard and all other plots are visually depicted (Figure 5(a)).

The scan occurs in increments of one residue at a time and can be either horizontal (from left to right) or vertical (from top to bottom). The size of the window can also be changed. Window movement is controlled by buttons that cause the scan to stop, to increment by one, or to decrement by one. In the main screen (left side of 5(a)), each subspace plot placed on the clipboard has a row consisting of bubbles showing the similarity of parameter behavior. Each bubble encodes the similarity between the flexibility values within this subspace plot and the flexibility values within another subspace plot during each scan. When covariances are small, bubbles are small

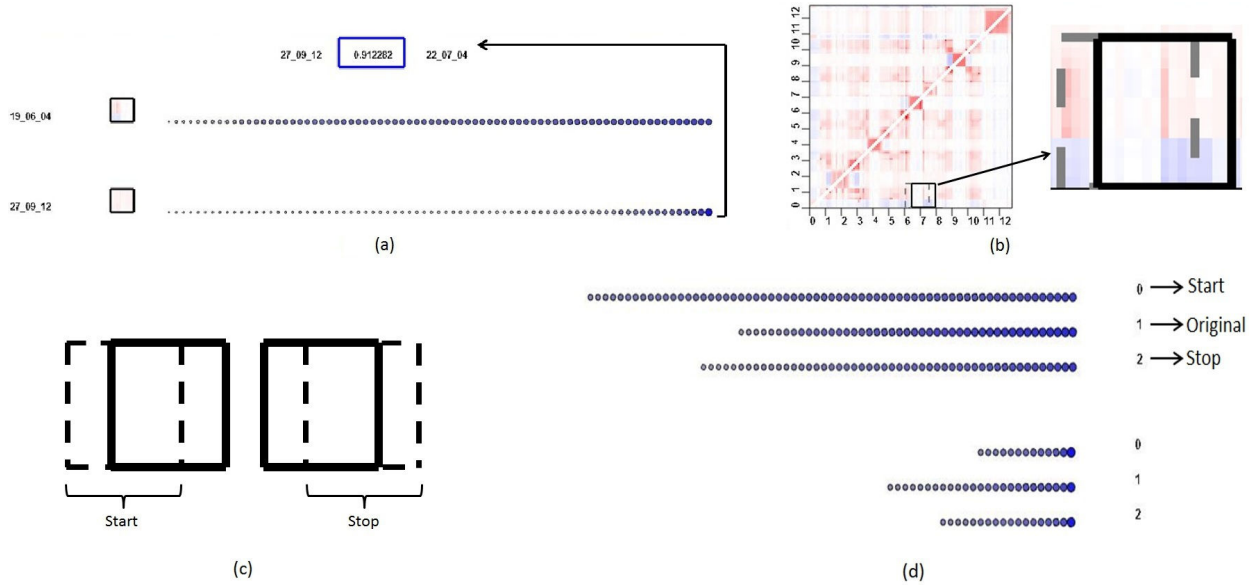

Figure 5: Scanning provides contextual information for the chosen subspace. (a) Sorted bubbles in each row trace the similarity between the subspace plot visualized on the far left and the corresponding area in the other plots. Detailed information is available for a bubble with a mouse-over. (b) The original subspace (dark box) and the current location of the sliding window (dotted box). (c) The starting and stopping positions of a horizontal scan. (d) The snapshot view for two subspaces being scanned in (a).

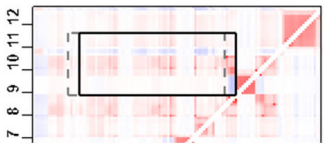

Figure 6: Window selecting an area of interest allows greater freedom when choosing subspaces.

and gray. Large covariances result in larger bubbles and deeper shades of blue. Bubbles in each row are sorted and collectively represent the similarity of the subspace plot on the far left to all other subspace plots occupying the same locations for other parameter sets. As the scan progresses, the bubble area and the alpha value are altered as similarities change. The bubbles and sliding window for a horizontal scan are shown in Figure 5(a) and 5(b).

When any bubble is selected by a mouse-over, the normalized distance is shown at the top of the screen within a rectangle that matches the corresponding bubble color (Figure 5(a)). On either side of the rectangle are the plot names for which that covariance was calculated. If the plot can not be evenly divided by the subspace length and width, a free-hand tool (Figure 6) is available by window selecting an area of interest.

5.2 Scan Snapshot

5.2.1 View Organization

The sliding subspaces were shown to the domain analysts. Although scientists could notice changes in the bubbles, they had difficulty describing the changes in similarity. Robertson et al. [17] found that static depictions of trends can be more effective analysis tools than animation. The snapshot view is a static depiction of bubble trends that improves on the original sliding subspace view by summarizing patterns difficult to follow during the animation. The system records the bubble sizes in each row at the beginning of the scan, at the original window, and at the end of the scan. These snapshots are automatically organized by the system during a scan and are available to the user after it.

Example 1.2: After a subspace of interest is identified, the two plots on the clipboard are examined using the snapshot view, one plot for each parameter set (see Figure 5(d)). The first one is shown in the top group of bubbles and the second one is shown in the bottom group of bubbles. For both plots, line 0 is the beginning of a scan, line 1 is the original window, and line 2 is the end of the scan. In this case, only bubbles having a normalized covariance greater than 0.5 are drawn. This helps filter subspace plots that are similar to the target and more clearly show the subspace plots that are less similar. In Figure 5(d), the top group of scans shows that the number of dissimilar subspace plots decreases as the original subspace is reached and then increases again as the sliding window passes through to the other side. ***For scientists, this case means that the top parameter set's influence on flexibility is more similar to the other parameter sets in the original subspace than in the surrounding subspaces.*** For the more consistent subspaces on the bottom, the opposite occurs. ***For scientists, this means that the bottom parameter set's influence on flexibility is similar to fewer other parameter sets in the original subspace than in the surrounding subspaces.*** Also clear in this view are general characteristics of the two parameter sets. The top group of scans in Figure 5(d) has more bubbles in each row than those in the bottom group. For the scanned subspaces, this indicates that the bottom subspace plot has more overall similarity to the rest of the data set than the top subspace plot. Scientists now have information to help them more precisely explore where parameter influences change in both the original subspace and the surrounding area.

6 Grouping Subspaces

6.1 Motivation

During development scientists communicated that after analyzing a subspace neighborhood, they would also like to narrow their analysis so that behavioral changes for a set of individual columns can be traced while systematically transitioning from one parameter set to another. The plot-carving view, better suited for larger subspaces, could not be easily adapted to this new requirement. In the case of a single residue column representing individual residue influence, selection is difficult because the subregions are so thin that consec-

utive regions are completely drawn over by the grid. Furthermore, the window slides would be too short to glean any useful information.

There is much work on visualizing clusters so that different cluster results can be compared [7], [16], [20]. We desired a solution capable of facilitating the analysis of clusters that changed in a cyclical manner. In other words, scientists should be able to cycle through parameter combinations in a manner that had mechanical or chemical meaning. This allows analysts to relate the change in behavior to parameter variations.

6.2 Tailed Transitions

To meet the new requirement, we developed a view called *tailed transitions* that integrates the Force-Directed Placement [10] algorithm. Alsakran et al. [1] use the algorithm to visualize streaming text documents where the closeness of node locations indicates similarity. Their system uses a force-based physical simulation of particles that represent data items. Specifically, the physical system exerts attracting and repelling forces that move similar particles closer and pull away dissimilar ones. The algorithm is repeated until the particles reach a state of equilibrium where their change in position is below a threshold.

In the simulation, the similarity between particles determines the direction and magnitude of the forces driving the particles. When the similarities between particles change, all particles move to new locations leading to a new visual layout. During the layout transformation, the transition behavior of the particles reflects the magnitude of similarity changes.

In this work, the force based method is extended and integrated so that the complete simulation is executed for a single parameter set plot. In [1] the visualization is updated after each text document particle enters the system at different time steps. The data set used here is not time dependent but the application is similar because the node positions are evolved from one layout to another. We adapt the visualization technique to discover the similarity changes among residue columns resulting from cycling parameter combinations. In particular, the layout for a single parameter set plot is not viewed until all of the columns, each of which is represented by a circle in the display, are fed to the simulation. This ensures that the order of the items do not affect the final layout. Column locations are evolved between parameter combinations by using the final location of data items for one parameter set as the beginning location for the start of the next simulation. Instead of animating layout changes, a static depiction as suggested by [17] is used. Static tails are drawn from the current location to the previous location. Long, dark tails indicate a large change in location. To track individual columns, the user can select an individual node by middle-mouse clicking. This makes the tail and node green so that tracking is easier. Once a data item is highlighted, the flexibility values in that column are drawn in the bottom left. Users can find particular items by entering the residue number into a search box. Figure 7 shows the tails tracing the change in similarity from successive parameter combinations.

Example 2: We now use an example to show how scientists are able to compare parameter behavior similarity for a single residue (an individual column) within a single plot as parameter combinations change. Scientists desire to establish the change in behavior for the energy found in the native state (natural or beginning shape) of local torsion interactions. Here, the interaction energy will range from 0.6 to 1.0 kcal/mol in increments of 0.1 kcal/mol. The other parameters, the average energy of a hydrogen bond and the entropy in the native state of a torsion angle, are held constant. The analyst is particularly curious about an area that ranges from residue 70 to 80. Data items are located and highlighted by entering their residue number in the search box and then middle-mouse clicking the node newly labeled. The transition from 0.6 to 0.7 kcal/mol results in the layout in Figure 7(a). There are few residues with a large change. However, the transition from 0.7 to 0.8 kcal/mol shows a large shift in the flexibility of residues 72, 74, and 79. The remaining transitions (0.8 to 0.9 and 0.9 to 1.0) result in small and diminishing shifts. ***For scientists, this means that as the interaction energy gradually increases, the greatest shifts in similarity for the residues of interest occurs when transitioning from 0.7 to 0.8 kcal/mol.*** Now, these three columns can be isolated and their data members examined in detail to gain a better understanding why they differ so much at that given change.

7 Expert Feedback

To evaluate the system, we met two of the domain analysts for a four hour evaluation session. The system was demonstrated to each of the analysts individually. During each demonstration, an analyst was given the opportunity to explore the data set with the system, ask questions, and give feedback. Following individual exploration and feedback, the two analysts jointly discussed the system. One analyst was surprised at the apparent variability in the set of sorted flexibility values. This spurred questions regarding model specifics and their influence on parameter sensitivity. He also stated that the snapshot view of the sliding subspaces can help classify subspaces for the purpose of filtering parameters and the tailed transisitons view can be helpful in examining temperature parameters in molecular dynamics simulations.

The other analyst gave a detailed example of an area other than allostery where the techniques in this tool can be helpful. Scientists often want to examine the similarities and dissimilarities of a specific part of a protein thought to be responsible for a specific function across many different species. By investigating these *functional domains*, scientists hope to identify and understand the parts responsible for small evolutionary differences that result in varying functionality. The analyst explained that the plot-carving view is well suited for isolating the sections of the protein thought to be involved and that the ability to identify, sort, and browse individual residues can be useful in more precisely locating the specific site on the protein. The stability of the site can be evaluated in the tailed transitions view. However, if it is concluded that the isolated residue is not the one responsbile for variation, the snapshot of the sliding subspaces can help expand the local search.

The domain analysts were eager to provide ideas about future directions. They stated that they would like more filtering and sorting options during analysis, especially by paramter type and value. Scientists also mentioned that they would like more robust capabilities for evaluting individual columns in the matrix. Furthermore, they noted that even incremental position changes by nodes in the tailed transitions view can accumulate across the data set and suggested that being able to visually compare the incremental changes to the total variability would help.

8 Conclusion

In this work, we have presented novel tools for investigating protein flexibility subspaces. The tools are a plot-carving view, sliding subspaces, and tailed transitions. The utility in meeting the defined high-level tasks were illustrated with detailed user scenarios and verified by domain experts. In the future, we plan to further the ability of scientists to coherently analyze residues that are not spatially adjacent and increase system scalability. We would also like to integrate a three-dimensional view so that when scientists find a single residue of interest, the residue can be placed into the context of a protein's true structure.

References

[1] J. Alsakran, Y. Chen, D. Luo, Y. Zhao, J. Yang, W. Dou, and S. Liu. Real-time visualization of streaming text with force-based dynamic system. *IEEE Computer Graphics and Applications (CG&A)*, 32(1):34–45, 2012.

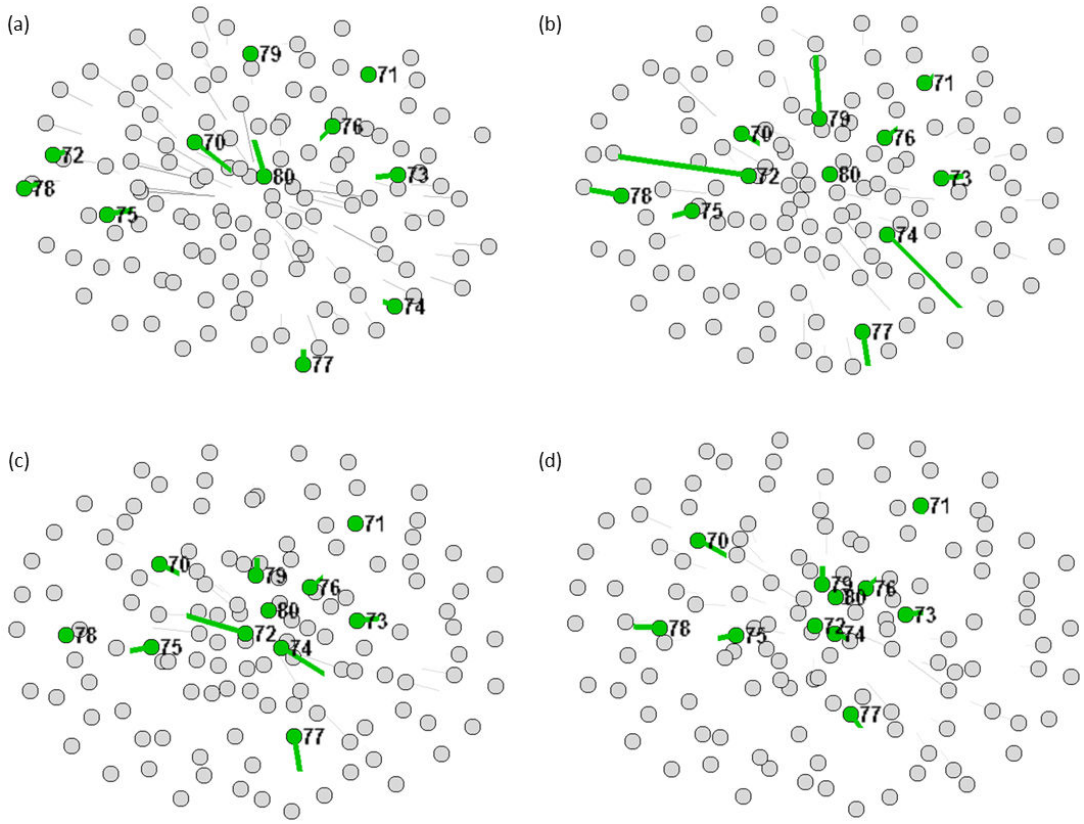

Figure 7: A series of transitions displaying residue column similarity for investigating local, native torsion energy. Changes in movement for selected items are marked with a green circle and tail. Transitions include (a) 0.6 to 0.7 kcal/mol, (b) 0.7 to 0.8 kcal/mol, (c) 0.8 to 0.9 kcal/mol, and (d) 0.9 to 1.0 kcal/mol. The greatest change within the highlighted items occurs for residues 72, 74, and 79 in (b) and for residues 72 and 74 in (c).

[2] I. Assent, R. Krieger, E. Muller, and T. Seidl. Visa: Visual subspace clustering analysis. *SIGKDD Explorations*, 9(2):5–12, 2007.

[3] S. Barlowe. *A Visual Analytics Approach to Feature Discovery and Subspace Exploration in Protein Flexibility Matrices*. PhD thesis, The University of North Carolina at Charlotte, 2011.

[4] S. Barlowe, Y. Liu, J. Yang, D. R. Livesay, D. Jacobs, J. Mottonen, and D. Verma. Wavemap: Interactively discovering features from protein flexibility matrices using wavelet-based visual analytics. *Computer Graphics Forum*, 30(3):1001–1010, 2011.

[5] S. Barlowe, T. Zhang, Y. Liu, J. Yang, and D. Jacobs. Multivariate visual explanation for high dimensional datasets. *IEEE Symposium on Visual Analytics Science and Technology*, pages 147–154, 2008.

[6] T. Bernas, E. K. Asem, J. P. Robinson, and B. Rajwa. Quadratic form: A robust metric for quantitative comparison of flow cytometric histograms. *Cytometry A*, 73(8):715–26, 2008.

[7] J. Chen, A. M. MacEachren, and D. J. Peuquet. Constructing overview + detail dendrogram-matrix views. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):889–896, 2009.

[8] B. J. Ferdosi, H. Buddelmeijer, S. Trager, M. H. Wilkinson, and J. B. Roerdink. Finding and visualizing relevant subspaces for clustering high-dimensional astronomical data using connected morphological operators. *Proc. IEEE Conference on Visual Analytics Science and Technology*, pages p. 35–42, October 2010.

[9] B. J. Ferdosi and J. B. Roerdink. Visualizing high-dimensional structures by dimension ordering and filtering using subspace analysis. *Eurographics/IEEE Symposium on Visualization*, 30(3):1121–1130, 2011.

[10] T. M. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Software - Practice and Experience*, 21(11):1129–1164, 1991.

[11] D. Guo. Coordinating computational and visual approaches for interactive feature selection and multivariate clustering. *Information Visualization*, 2(4):232–246, 2003.

[12] Z. Guo, M. O. Ward, and E. A. Rundensteiner. Model space visualization for multivariate linear trend discovery. *Proc. IEEE Conference on Visual Analytics Science and Technology*, pages 75–82, 2009.

[13] D. J. Jacobs, D. R. Livesay, J. M. Mottonen, O. K. Vorov, A. Y. Istomin, and D. Verma. Ensemble properties of network rigidity reveal allosteric mechanisms. *Methods in Molecular Biology*, 796:279 – 304, 2012.

[14] V. Kondekar, V. Kolkure, G. Sodal, and J. Mudegaonkar. Image retrieval techniques based on image features: a state of art approach for cbir. *International Conference & Workshop on Emerging Trends in Technology*, 2010:998–999, 2010.

[15] J. Mottonen, D. Jacobs, and D. Livesay. Allosteric response is both conserved and variable across three chey orthologs. *Biophysical Journal*, 99(7):2245–54, 2010.

[16] L. Nowell, E. Hetzler, and T. Tanasse. Change blindness in information visualization: A case study. *Proceedings of the IEEE Symposium on Information Visualization*, pages 15–22, 2001.

[17] G. Robertson, R. Fernandez, D. Fisher, B. Lee, and J. Stasko. Effectiveness of animation in trend visualization. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1325–1332, 2008.

[18] Y. Rubner, C. Thomasi, and L. J. Guibas. The earth movers distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[19] F. R. Salsbury. Molecular dynamics simulations of protein dynamics and their relevance to drug discovery. *Current Opinion in Pharmacology*, 10(6):738 – 44, 2010.

[20] J. Seo and B. Shneiderman. Interactively exploring hierarchical clustering results. *IEEE Computer*, 35(7):80–86, 2002.