Visualization of Lattice-Based Protein Folding Simulations

Sebastian Pötzsch[1], Gerik Scheuermann[1], Michael T. Wolfinger[2],
Christoph Flamm[1,2], Peter F. Stadler[1],
[1]Department of Computer Science - University of Leipzig,
[2]Institute of Theoretical Chemistry - University of Vienna
mai00cmi@studserv.uni-leipzig.de, scheuermann@informatik.uni-leipzig.de, mtw@tbi.univie.ac.at,
xtof@tbi.univie.ac.at, studla@bioinf.uni-leipzig.de

Abstract

Analysis of the spatial structure of proteins including folding processes is a challenge for modern bioinformatics. Due to limited experimental access to folding processes, computer simulations are a standard approach. Since realistic continuous (all-atom) simulations are far too expensive, lattice based protein folding simulations are a common coarse-graining. In this paper, we present a visualization tool for lattice based protein folding simulations. The system is based on Shneiderman's mantra "Overview first, zoom and filter, details on demand" and uses a collection of information visualization techniques including multiple views, focus+context and table lenses which have been tailored towards our data. We demonstrate the potential of information visualization techniques for providing insight into such simulations.

Keywords— **Information visualization, multiple views, overview+detail, design guidelines, focus+context.**

1 Introduction

One of the longstanding unsolved computational problems in bioscience is the folding of biopolymers i.e. the transition of the molecule from an inactive denatured structure to the fully functional three-dimensional structure. It is well known that a protein folds into its native three-dimensional structure spontaneously under certain physiological conditions. During this process the chemical information stored in the amino acid sequence is translated into a scaffold of non-covalent interactions which stabilizes the three-dimensional structure. Mutations in the protein sequence may cause severe changes of the three-dimensional structure which in turn has a crucial impact onto the function of the protein.

To gain deeper insight into the generic properties of the protein folding process coarse-grained models like the lattice protein model are of particular interest since they open up a feasible route in terms of computational resources to study this important problem. Understanding the sequence to structure map of proteins would open up new horizons in fields like drug design or nanotechnology and may give detailed insights into the mechanisms of protein related diseases such as Alzheimer, BSE and Parkinson.

A common approach to coarse grained protein models is to reduce the number of possible interactions between chemically different amino acid by lumping the 20 amino acids together into two classes, a hydrophilic (H) and a polar (P) one. The rational behind this clustering of the amino acids into two types comes from the observation that the core of natural globular proteins is predominantly formed by amino acids which can be classified chemically as hydrophobic, while the residues composing the surface, the part of the protein that comes into close contact with the surrounding solvent, are almost exclusively polar ones.

The HP-type-models introduced by Dill and Lau [10] use this simplification and measure energy contributions only between adjacent amino acids. To simplify the calculation of the neighborhood these type of model discretizes space to a particular lattice where amino acids are allowed to occupy lattice sites only.

Although these models are not applicable for predicting the native structure for real proteins, they give insight into general properties of protein folding. Berger and Leighton [3] have shown that even with these simplifications protein folding on the HP-model is NP-complete, illustrating the complex nature of the problem.

We here present an information visualization tool that supports the analysis and exploration of huge data sets and leads to a basic understanding of these simulations. We will use concepts like multiple views, overview+detail and focus+context to prove how these techniques can amplify cognition and solve visualization tasks.

2 Related Work

Many visualization tools are designed according to Shneiderman's mantra "Overview first, zoom and filter,

Proceedings of the Information Visualization (IV'06)
0-7695-2602-0/06 $20.00 © 2006 **IEEE**

details on demand.” [15]. The problems related to large data sets and visualization with limited display size are well known and addressed in a series of works (see [5] for an overview). Several concepts exist how to preserve an overview while exploring focused samples out of a huge data set. One concept that is used in Seesoft [6] and many other applications (see [14, 9, 4]) is to show a detail view for selected data samples separated from a visual overview of the whole content. For example, Seesoft visualizes program code presenting an overview of the whole file where the lines are presented by simple lines and uses a detail view for a snippet to show in a readable manner.

Other methods give an overview and details in just one view using distortions [8, 12, 13]. These techniques are called focus plus context visualizations and allow users to view selected data samples in additional detail without requiring a second window. For example, the Document Lens application [13] visualizes all pages of a document. The pages are laid out in rows around a focal area and the user can zoom in on pages to make them readable using a rectangular area, and pan to branch other pages into focus. The pages not in focus are distorted to fit the area outside of the rectangular area. In other works [8, 12], a degree of interest function is used to map screen space to each data sample. The remaining samples share the rest of the screen space. Therefore, the result is a distorted representation according to the degree of interest. A taxonomy of such distortion-oriented presentation techniques is given by Apperley [1].

For multiple views of the selected data samples, there are several guidelines presented for design decisions addressing the layout and coordination mechanisms by Baldonado [2]. A taxonomy of linking techniques and coordination of multiple views is given by Shneiderman [16].

3 Protein Folding Simulation

In this section we will briefly describe the underlying data. A protein is composed of unbranched chains of amino acids connected by chemical bonds. There are 20 different possible amino acids occurring in arbitrary number and sequence in different proteins. For biological processes, the spatial arrangement of this chain is of equal importance as the chain itself since the spatial structure crucially determines the function of the protein. The process of building or changing the spatial structure is called protein folding. It is triggered by the free energy associated with each structure. A stable structure has a lower energy than all other structures that can be built by changes due to thermal movement of the atoms in the molecule. Unfortunately, realistic simulations using all atoms are far too expensive. A common coarse-graining method are lattice-based protein folding simulations. In the HP-model [10], amino acids are modeled as points sitting on a lattice and the bonds as edges combining neighboring points. All amino acids are divided into two classes, a hydrophobic and a hydrophilic class since this distinction has the strongest effect on the structure. The structure is described by a self-avoiding walk of the chain on the lattice and the folding is modeled with a small set of basic transformations of these walks.

The Pinfold program, based on the algorithm presented in [7], is a rejection-less Monte Carlo method to simulate the kinetic folding of lattice heteropolymers as a homogeneous, continuous-time Markov chain using the simplified HP-model [10]. Structures are represented as strings of letters which correspond to relative moves on the lattice. A variety of standard lattices are available in two and three dimensions. The objective is to analyze the folding kinetics by means of exploring the energy landscape at elementary step resolution. Here, elementary steps, describing the transformation between different structures correspond to pivot moves [17] (see Figure 1).

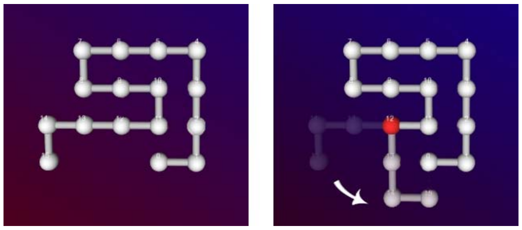
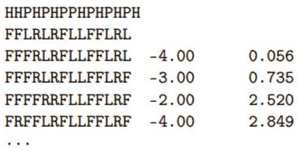


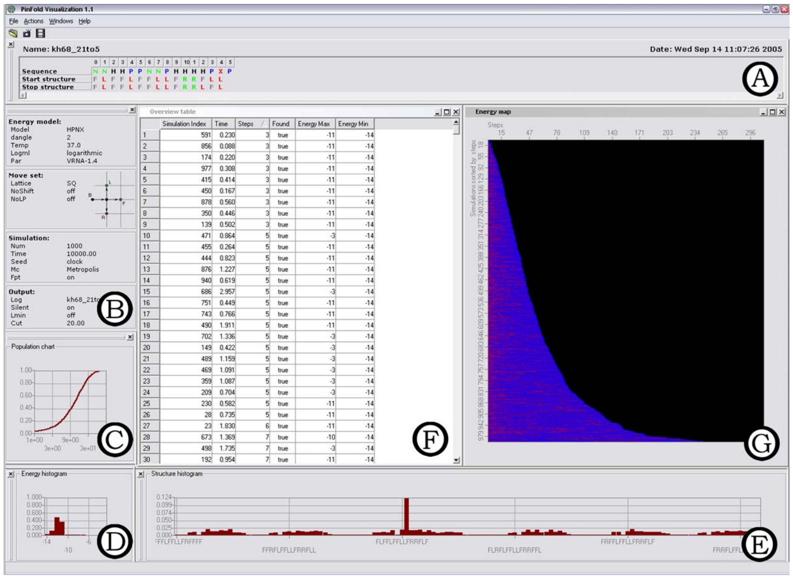
Figure 1: Transformation from one structure into another by a pivot move (rotation around the red bead).

Within the Pinfold framework, pivot moves are particularly easy to implement, since they correspond to an exchange by one letter to another one (point mutation) in the relative move string. This coarse-graining is suitable to simulate the kinetics of protein folding. Starting from a given structure, Pinfold calculates an ensemble of folding trajectories, which are time series of structural changes together with their energies (see Figure 2). From these, important characteristics and patterns of the folding process can be extracted using the visualization tool presented here which are otherwise hard to find. Since a stochastic method is used, thousands of simulations must be analyzed to get statistically significant results.



```
HHPHPHPPHPHPHPH
FFLRLRFLLFFLRL
FFFRLRFLLFFLRL  -4.00      0.056
FFFRLRFLLFFLRF  -3.00      0.735
FFFFRRFLLFFLRF  -2.00      2.520
FRFFLRFLLFFLRF  -4.00      2.849
...
```

Figure 2: Sample trajectory of Pinfold with amino acid chain (in HP-Model), start structure and table with structures, free energy and elapsed time



Name: kh68_21to5

Date: Wed Sep 14 11:07:26 2005

	0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5
Sequence	N	N	H	H	P	P	N	N	P	H	H	H	H	P	X	P
Start structure	F	L	F	F	L	F	F	L	L	F	R	R	F	L	L	
Stop structure	F	L	F	F	L	F	F	L	L	F	R	R	L	F	L	

Energy model:

Model	HPNX
dangle	2
Temp	37.0
Logml	logarithmic
Par	VRNA-1.4

Move set:

Lattice	SQ
NoShift	off
NoLP	off

Simulation:

Num	1000
Time	10000.00
Seed	clock
Mc	Metropolis
Fpt	on

Output:

Log	kh68_21to
Silent	on
Lmin	off
Cut	20.00

	Simulation Index	Time	Steps	Found	Energy Max	Energy Min
1	591	0.230	3	true	-11	-14
2	856	0.088	3	true	-11	-14
3	174	0.220	3	true	-11	-14
4	977	0.308	3	true	-11	-14
5	415	0.414	3	true	-11	-14
6	450	0.167	3	true	-11	-14
7	878	0.560	3	true	-11	-14
8	350	0.446	3	true	-11	-14
9	139	0.502	3	true	-11	-14
10	471	0.864	5	true	-3	-14
11	455	0.264	5	true	-11	-14
12	444	0.823	5	true	-11	-14
13	876	1.227	5	true	-11	-14
14	940	0.619	5	true	-11	-14
15	686	2.957	5	true	-3	-14
16	751	0.449	5	true	-11	-14
17	743	0.766	5	true	-11	-14
18	490	1.911	5	true	-11	-14
19	702	1.336	5	true	-3	-14
20	149	0.422	5	true	-3	-14
21	489	1.159	5	true	-3	-14
22	469	1.091	5	true	-3	-14
23	359	1.087	5	true	-3	-14
24	209	0.704	5	true	-3	-14
25	230	0.582	5	true	-11	-14
26	28	0.735	5	true	-11	-14
27	23	1.830	6	true	-11	-14
28	673	1.369	7	true	-10	-14
29	498	1.735	7	true	-3	-14
30	192	0.954	7	true	-11	-14

Figure 3: The overview after loading the data

4 The Task

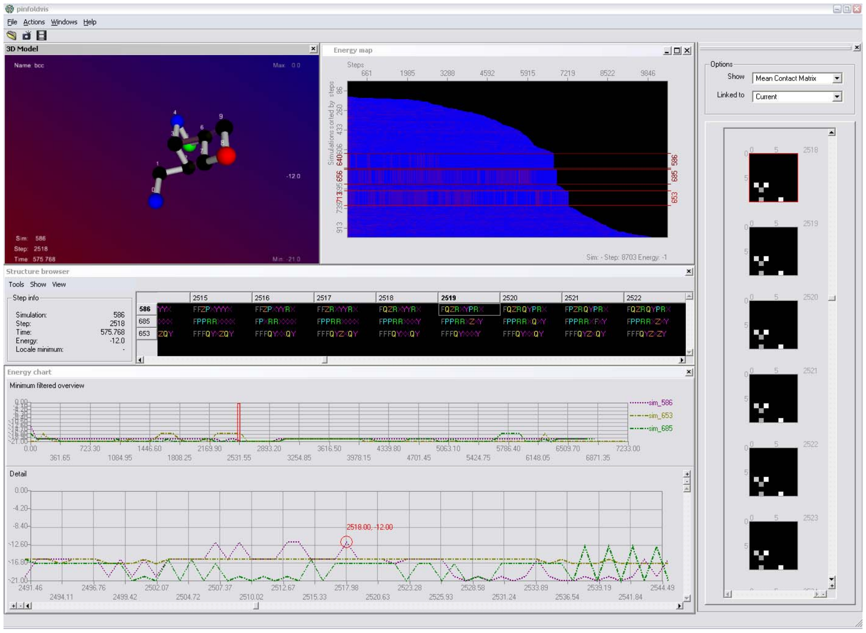
The output of Pinfold contains several thousand simulations and several hundred thousands of simulations steps. These data sets are impossible to explore without a visualization tool. Our task was to provide a tool that allows to analyze the data and emphasize relationships in the simulations. The hope is to uncover regularities which help to understand the folding process. Another aspect is to find heuristics to stop simulations that would take too long to find a stop structure.

5 Overview Visualization

For the design of our visualization tool, we follow the frequently quoted mantra by Ben Shneiderman [15] ”Overview first, zoom and filter, details on demand.” After the user has loaded the data we show an overview for a better orientation. We use several views with global information over the simulations (see Figure 3). The view (A) shows the date of the simulation, the name of the protein, the protein sequence, the start structure and the stop stucture. View (B) displays the parameters chosen for the simulations (e.g. the used lattice, energy model, number of simulations, time, etc.). In view (C) we show the population chart. This curve shows how many simulations out of all simulations found the stop structure at a time. The shape of the curve is interesting for the global behavior of the simulations and can give some hints of metastable structures that can occur in the folding from the start structure to the stop structure. The two views at the bottom of Figure 3 show the distribution of specific energy values (D) respectively certain structures (E). They allow to recognize structures or energy values which appear very often and therefore could be interesting.

5.1 Filtering and Zooming

The next step according to Shneiderman’s mantra is to provide the possibility to pick out some simulations for a comparison or a more detailed exploration. Thus, we provide two views (see Figure 3 (F)-(G)) in which the user can search for significant features. The first view (F) is a normal data table. Simulations are shown in rows and columns display some characteristics (e.g. number of steps, duration, an indicator if the stop structure was found, energy minimum and energy maximum). The second view (G) is an energy map in which the energy of every simulation step of every simulation is represented by a color. Blue colors indicate low energies and red colors indicate high energies. The simulations are sorted by their duration. Since we are dealing with more than ten thousand simulation steps and more than thousand simulations using scrolling and zooming to discover this map would be very inappropriate. Therefore, we use a focus+context technique that is based on the table lens by Rao [12]. This technique is useful to display an overview and details of the focus in just one view. Like Rao we’re using two degrees of interest functions, one for a horizontal focus and one for a vertical focus. This method allows us to compare different simulations and different simulation steps at a glance (see Figure 4).



Step info

Simulation: 586
Step: 2518
Time: 575.768
Energy: -12.0
Locale minimum: -

Options

Show: Mean Contact Matrix
Linked to: Current

Figure 5: The multiple views

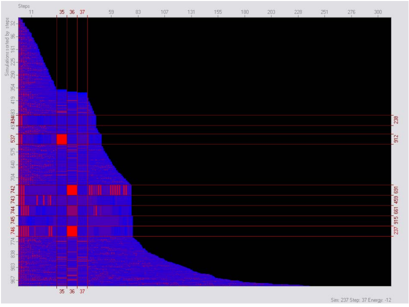


Figure 4: The energy map with focused simulations and steps

5.2 Detail Views

After the user has chosen some simulations either in the data table or in the energy map he can invoke more detailed views that show different aspects of the data (see Figure 5). This technique is often called multiple view technique. We followed the guidelines of Baldonado [2] for the design of multiple views. Because we are dealing with 3D structures we have a 3d viewer that shows the structures (see Figure 6). In this viewer different color mappings are possible. For example, one color mapping shows the amino acids according to the used energy model (HP/HPNX). Another one uses a mapping that emphasizes contact changes by highlighting amino acids that formed a new contact (green) or a lost contact (red). Another representation shows the pivot point of the last transformation. The background of the 3d viewer is colored according to the energy of the structure so that a fast browsing through a simulation leads to a significant flickering if the energies differ a lot.

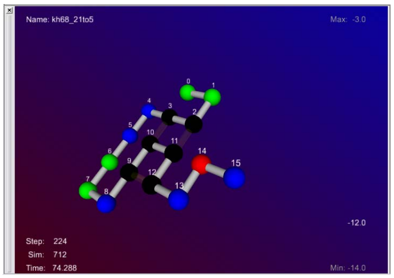


Figure 6: The 3d viewer

Also of special interest to biologists is the curve of the energy in a simulation. Thus, a chart viewer showing the energy curves of selected simulations is very helpful to compare simulations according to energies of single steps. We are using a overview+detail technique where two views are used. One view shows the complete energy curve which is down sampled due to limit screen space. The other view shows the curve in more detail for a snippet of the simulation. The snippet is indicated by a red rectangle in the overview (see Figure 7).

Another very interesting aspect besides the energies are the contacts or positions of the amino acids in the structure. Therefore, a matrix viewer addresses the relationships between several amino acids (see Figure 8).

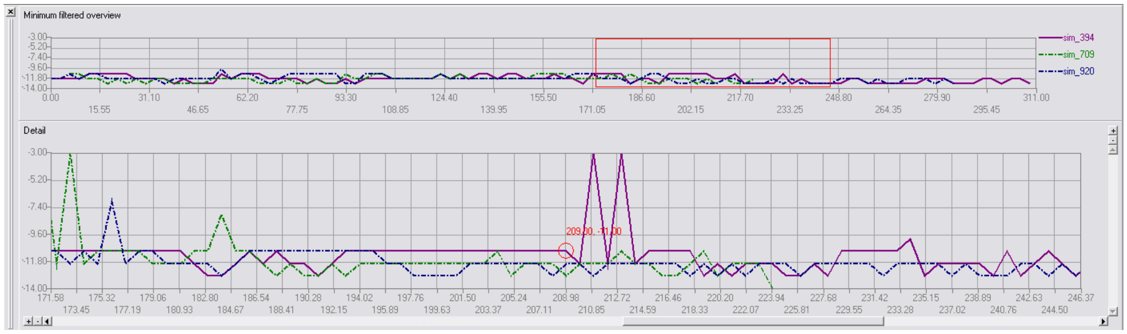
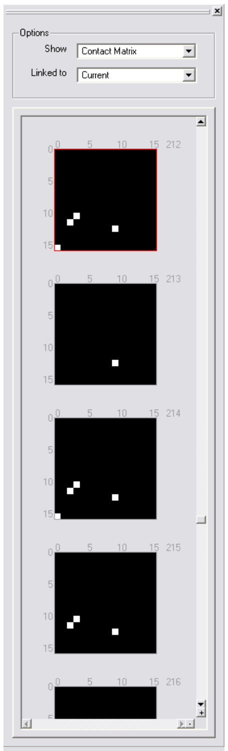


Figure 7: The energy curves (red rectangle shows snippet)


Options

Show: Contact Matrix

Linked to: Current

Figure 8: The matrix browser

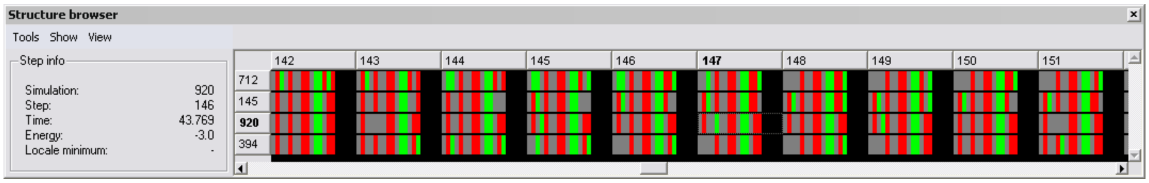
This viewer shows different matrices depending on the structure, e.g. a contact matrix indicating bondings between amino acids. In this matrix, contact patterns appearing very frequently stick out and indicate parts of the structure that are significant in a simulation. Another choice is the difference matrix between two adjacent simulation steps. The idea is that large changes in the contact matrices can be found easily. It is also possible to look at the positions of the amino acids in a distance matrix emphasizing patterns in their spatial relationships. The last matrix that can be used to analyze the simulation steps is the variance of the distance matrix. This matrix provides hints at parts of the structure that are stable in the positions to each other and hence emphasizes parts that are very volatile. To compare selected simulations based on the structures, we provide a structure browser (see Figure 9). The structures are shown in a string representation in which every character stands for a relative move on the lattice. So the structures are described like a path description. We allow different representations of these strings. The first one shows the structure as strings with different colors for the relative moves and the second shows small colored bars for a relative move. The idea is to see patterns easier due to the coloring. These patterns can highlight parts of the structures that don't change very often and therefore seem to be important to lead into a low energy. It is also possible to align two simulations by a Needleman-Wunsch algorithm [11] to compare them and analyze their similarities.

All our detail views are linked by navigational linking so that picking out and clicking at one structure is passed to all the other views and vice versa. So the 3d structure, the belonging matrix and the position in the energy curve is updated. The linking makes it easy to emphasize the relationships between the different aspects of a protein structure (e.g. looking at the structure of a interesting matrix or point in the energy curves).

6 Results

The first result gained with our tool was the detection of implementation errors in the simulation program causing inconsistent data. The first mistake was an error in the implementation of the face centered cubic lattice in which different letters for relative moves described the same structure. The second error caused simulations to terminate after just one simulation step was calculated.

The typical user scenario seems to be the following pattern. After a study of the overview (Figure 3) especially the population chart (Figure 3(C)) and energy map (Figure 3(G)), the user looks for salient simulations with significant energy patterns in the energy map. Using the table lens technique and checking the color coding (Figure 4), such salient simulations attract the user's interest. The im-



Step info

Simulation:	920
Step:	146
Time:	43.769
Energy:	-3.0
Locale minimum:	-

Figure 9: The structure browser

mediate response is a look at the energies curves (Figure 7) and an activation of the 3D viewer (Figure 6). Since the three-dimensional lattices are not all very intuitive, the chemists and biologist in our team use the 3D viewer intensively to study the pivot moves and try to get an impression of the changes. Due to the linked view concept, it is simple to select high energy jumps in the simulation and to get an immediate visible response of its structural meaning. After some time, the structure browser and the contact matrix viewer (Figure 8) are used to get more structural information about the folding process. The intensive study of simulations is repeated quite often and has already provided understanding of the simulated folding processes. We have found common patterns in the contact matrices indicating stable parts in the structure using the matrix browser. The program is currently used to study more simulations runs and we expect deeper biological results in the future.

Conclusions

We implemented a tool based on state of the art visualization techniques to support access to the `Pinfold` output. It was shown that Shneiderman's mantra is a good guide for this kind of data. By implementing a focus+context technique to support filtering of large data sets, we guide the user to explore significant simulations. The revealing of relationships between spatial structures, energies and patterns in simple matrices by the application of multiple views was also shown. The success of our program so far is the revealing of simulation errors, the access to the `Pinfold` output and the concentration on significant simulations. We are currently, in collaboration with the University of Vienna, waiting for biological results based on our program.

Acknowledgements

This work was supported in part by the EMBIO project in FP-6 (`http://www-embio.ch.cam.ac.uk/`).

References

[1] Mark D. Apperley and Y. K. Leung. A review and taxonomy of distortion-oriented presentation techniques. *ACM Transactions on Computer-Human Interaction*, 1(2):126–160, June 1994.

[2] Michelle Q. Wang Baldonado, Allison Woodruff, and Allan Kuchinsky. Guidelines for using multiple views in information visualization. In *Advanced Visual Interfaces*, pages 110–119, 2000.

[3] Bonnie Berger and Frank Thomson Leighton. Protein folding in the hydrophobic-hydrophilic(HP) model is NP-complete. *Journal of Computational Biology*, 5(1):27–40, 1998.

[4] Donald Byrd. A scrollbar-based visualization for document navigation. *CoRR*, cs.IR/9902028, 1999.

[5] Stuart K. Card, Jock D. Mackinlay, and Ben Shneiderman, editors. *Readings in Information Visualization — Using Vision to Think*. Morgan Kaufmann, 1999.

[6] Stephen G. Eick, Joseph L. Steffen, and Eric E. Sumner Jr. Seesoft—A tool for visualizing line oriented software statistics. *IEEE Transactions on Software Engineering*, 18(11):957–968, November 1992.

[7] C. Flamm, W. Fontana, I. L. Hofacker, and P. Schuster. RNA folding kinetics at elementary step resolution. *RNA*, 6:325–338, 2000.

[8] G. W. Furnas. The FISHEYE view: A new look at structured files. Technical Memorandum #81-11221-9, Bell Laboratories, Murray Hill, New Jersey 07974, U.S.A., 12 October 1981.

[9] Jamey Graham. The reader's helper: A personalized document reading environment. In *CHI*, pages 481–488, 1999.

[10] Kit Fun Lau and Ken A Dill. A lattice statistical mechanics model of the conformational and sequence spaces of proteins macromolecules. *Macromolecules*, 22:3986–3997, 1989.

[11] Saul B. Needleman and Christian D. Wunsch. A general method applicable to the search for similarity in the amino acid sequences of two proteins. *Journal of Molecular Biology*, 48:443–453, 1970.

[12] Ramana Rao and Stuart K. Card. The table lens: Merging graphical representations in an interactive focus+context visualization for tabular information. In *Proceedings CHI 94*, pages 318–322. ACM, 1994.

[13] George G. Robertson and Jock D. Mackinlay. The document lens. In *Proceedings of the 6th Annual Symposium on User Interface Software and Technology*, pages 101–108, New York, NY, USA, November 1993. ACM Press.

[14] Alex Shah and Tony Darugar. Creating high performance web applications using tcl, display templates, XML, and database content, September 15 1998.

[15] Ben Shneiderman. *Designing the User Interface*. Addison Wesley Longman, third edition, 1998.

[16] Ben Shneiderman and Chris North. A taxonomy of multiple window coordinations, 1997.

[17] Alan D. Sokal and N. Madras. The pivot algorithm: A highly efficient monte carlo method for the self-avoiding walk. *Journal of Statistical Physics*, 50:109–189, 1988.