

Saliency Guided Summarization of Molecular Dynamics Simulations

Robert Patro¹, Cheuk Yiu Ip¹, and Amitabh Varshney¹

¹ University of Maryland, College Park MD 20742, USA
{rob,ipcy,varshney}@cs.umd.edu

Abstract

We present a novel method to measure saliency in molecular dynamics simulation data. This saliency measure is based on a multiscale center-surround mechanism, which is fast and efficient to compute. We explore the use of the saliency function to guide the selection of representative and anomalous timesteps for summarization of simulations. To this end, we also introduce a multiscale keyframe selection procedure which automatically provides keyframes representing the simulation at varying levels of coarseness. We compare our saliency guided keyframe approach against other methods, and show that it consistently selects superior keyframes as measured by their predictive power in reconstructing the simulation.

1998 ACM Subject Classification J.3 Life and Medical Sciences

Keywords and phrases Molecular Dynamics, Saliency, Simulation

Digital Object Identifier 10.4230/DFU.SciViz.2010.321

1 Introduction

Molecular dynamics trajectories play a vital role in enhancing our understanding of the building blocks of life at the nanoscale. A number of recent advances in modeling and simulation of proteins and nucleic acids continue to provide us with novel insights into the relationship between the form and function of these dynamic biological nanomachines. In our efforts to simulate ever-more accurate models of physics and chemistry, such simulations out of necessity have to occur over very small time scales, typically femtoseconds. However, the major molecular conformational changes of interest typically occur over timescales ranging from a few microseconds to seconds. This difference in simulation timescales is being bridged by novel algorithmic approximations, advances in hardware, as well as by simply running longer time-scale simulations facilitated by larger storage capacities of modern computer systems [9].

However, just because we now have the ability to simulate exceedingly long timescale molecular dynamics simulations, does not necessarily mean that we are better equipped to gain visual insights using such simulation datasets. Since the capabilities of the human visual system remain unchanged and the bandwidth into the human cognitive machinery remains constant, we have now reached a stage where the current generation simulation datasets can easily overwhelm the limits of human comprehension. In real world, the human visual system deals with the glut of information coming at it from the world by focusing retinal hardware and attention on what is most important, or salient. The challenge of visual presentation and analysis of very large datasets compels us to re-examine not just how to present data, but what data to present. In this paper we discuss some of our recent research that deals with how to effectively summarize large molecular dynamics trajectories using ideas inspired by the visual saliency mechanism of the human visual system.



© R. Patro, C.Y. Ip, and A. Varshney;
licensed under Creative Commons License NC-ND

Scientific Visualization: Advanced Concepts.

Editor: Hans Hagen; pp. 321–335



Dagstuhl Publishing

Schloss Dagstuhl – Leibniz Center for Informatics (Germany)

The main contributions of this paper are:

- We present a multiscale saliency operator for molecular dynamics trajectories that is successful at identifying the most salient time steps of a molecular dynamics simulation.
- We show how one can find the most representative frames of a molecular dynamics trajectory by simply inverting the multiscale saliency operator.
- We quantitatively show the benefits of using our multiscale saliency operator to summarize molecular dynamics trajectories compared with other methods such as a random scheme or the Douglas-Peucker scheme.
- We validate our methods using several real-world examples of long time scale molecular dynamics trajectories.

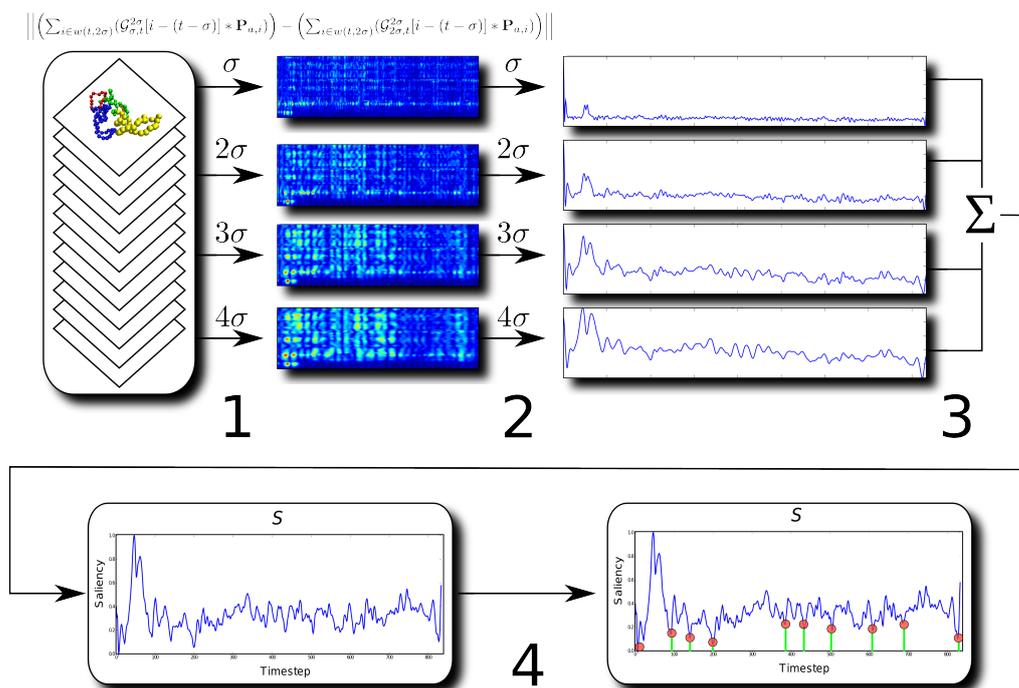
We believe that the research directions that we have identified in this paper are early-stage efforts that will hopefully spark a number of follow-on methods to automatically or semi-automatically identify and visualize salient features, events, and trends in very large-scale time-varying datasets. Our methods, in turn, build upon the seminal work of several other prominent researchers in the field. We give a summary of the related research in the next section.

2 Background and Related Work

The last few years have seen a growing interest in summarization of large-time varying datasets. Perhaps the greatest amount of research has been in abstraction [14] and summarization of videos [3]. However, much of the research on video summarization is not directly applicable to summarization of 3D datasets. In graphics, very interesting work has been done in summarization of articulated characters [2] as well as their compression [11, 1]. In volume visualization, Silver and Wang [12] have used the framework of template matching to identify key features such as reconnection events in large time-varying 3D volumes, such as those arising from computational fluid dynamics (CFD) simulations.

Almost all the previous work in characterization of time-varying datasets has been with either articulated skeletal human models or time-varying volume datasets. Although these methods provide helpful insights, they do not directly carry over for use in molecular dynamics simulations. This is because, unlike molecular dynamics simulations the movements in character animation are purposive, described by a set of continuously changing joint angles. Thus an event of interest in character animation can be detected by a change of such movements. Molecular dynamics simulations are characterized by a variety of motions at multiple scales. There are fine-scale Brownian motions and larger-scale conformational changes. To handle this challenge, we define a multiscale saliency operator that works with several different-sized sliding windows.

Ideally, we would like to identify the most important time steps in large time-varying molecular dynamics simulations for the purposes of summarization, fast-previewing, indexing, and further analysis. We build upon the ideas of image saliency by Itti *et al.* [8] and mesh saliency by Lee *et al.* [10]. They use a center-surround operator to identify the uniqueness of a pixel or a vertex with respect to its neighborhood. In our approach, we define the importance of a timestep by its difference from its neighbors, both forward and backward in time, over multiple scales. Specifically, we analyze a difference of Gaussian weighted average positions centered around each timestep for several different scales. Subsequently, we combine this information into a single multiscale saliency function \mathcal{S} , and present a multiscale keyframe selection procedure to obtain representative (or conversely, anomalous) frames based upon this function. Further details of our approach are in Sections 3 and 4.



■ **Figure 1** This figure gives an overview of our salient timestep selection procedure. In step 1, the per-atom saliency measure is taken at multiple scales. Step 2 then composites these per-atom saliency functions into a single per-timestep saliency function for each scale. In step 3, the separate single-scale saliency functions are combined into a single multiscale saliency function. Finally, in step 4, the multiscale keyframe selection (MSKS) procedure is used to select representative timesteps from the simulation using the multiscale saliency function.

3 Multiscale Saliency for Molecular Dynamics Trajectories

The notion of saliency is general. Indeed, even in the particular domain of molecular dynamics simulations, one might reasonably suggest many methods by which to compute saliency. In order to focus the scope of this work, we shall consider saliency on a per-timestep basis. In section 4, we consider the representation of simulation data through the selection of key timesteps. Thus, we are motivated to define our saliency measure on a per-timestep basis. Inspired by the saliency mechanism of the human visual system, we have decided to formulate our saliency operator for molecular dynamics trajectories to be multiscale and to make use of a center-surround mechanism [8].

Even among those measures adhering to these criteria, there are numerous possible definitions of per-timestep saliency. However, since the notion of saliency in the domain of molecular dynamics simulations has not been well explored, we shall introduce a simple but straightforward definition, which we have found to be very effective. An overview of our method is given in figure 1.

3.1 Saliency Definition

In order to define our notion of saliency, it will be necessary to introduce some notation. Our main analysis will be on an order-3 tensor \mathbf{P} , containing the position of each atom for every timestep of the simulation data. For a simulation with n timesteps and m atoms, \mathbf{P}

will be an $m \times n \times 3$ tensor, such that $\mathbf{P}_{a,t,-} = p_{a,t}^{\rightarrow}$ is the 3-vector containing the Cartesian coordinates of atom a at timestep t . For the sake of brevity, we shall write $\mathbf{P}_{a,t,-}$ as $\mathbf{P}_{a,t}$. It will also be useful to index ranges of \mathbf{P} . The expression $\mathbf{P}_{I,J}$ will be used to denote a block of values from \mathbf{P} spanning atoms $I = [i_1, i_2, \dots]$, timesteps $J = [j_1, j_2, \dots]$, and all spatial coordinates.

Further, it will be useful to define the notion of a selection window. Let $w(i, \sigma) = [i - \frac{\sigma}{2}, \dots, i + \frac{\sigma}{2}]$ denote the selection window of size $\sigma + 1$ centered about i , where σ is assumed to be an even integer. As these selection windows will be used to address our position tensor \mathbf{P} , we must be careful to assure they have a valid definition for each timestep i . This is achieved simply by “reflecting” the simulation data about the first and last frame to handle the respective border cases.

Finally, we shall make use of the notion of a discretely sampled Normal distribution. We denote by $N(i, \sigma)$, the normal distribution with mean i and variance σ , and by $N(i, \sigma)(j)$, the evaluation of $N(i, \sigma)$ at j . Then, we define $\mathcal{G}_{\sigma,i}^k = [N(i, \sigma)(i - \frac{k}{2}), \dots, N(i, \sigma)(i), \dots, N(i, \sigma)(i + \frac{k}{2})]$ as the set containing k values, each value obtained via the evaluation of the appropriately parametrized normal distribution at a given location. $\mathcal{G}_{\sigma,i}^k$ is symmetric about its center, which is at the $\frac{k}{2}$ th entry, $\mathcal{G}_{\sigma,i}^k[\frac{k}{2}]$.

3.1.1 Per-atom Saliency

In practice, we define our saliency field for each atom and for each timestep over a number of scales. To simplify the exposition, we will first consider only a single scale σ . Let us consider computing the saliency of atom a at timestep t ; it is given by the following equation:

$$\mathcal{S}_\sigma[a, t] = \left\| \left(\sum_{i \in w(t, 2\sigma)} (\mathcal{G}_{\sigma,t}^{2\sigma}[j] * \mathbf{P}_{a,i}) \right) - \left(\sum_{i \in w(t, \sigma)} (\mathcal{G}_{\sigma,t}^\sigma[j] * \mathbf{P}_{a,i}) \right) \right\| \quad (1)$$

Where $j = i - (t - \sigma)$ is used as the local index into the discretely sampled Normal distributions. Essentially, we consider the Gaussian weighted average of the atom’s position at scales σ and 2σ , and define the saliency as the norm of the difference. In this formulation, we consider σ as the “center” scale and 2σ as the “surround” scale. The greater the difference between the center and the surround, the higher the saliency at the point of evaluation.

3.1.2 Per-timestep Saliency

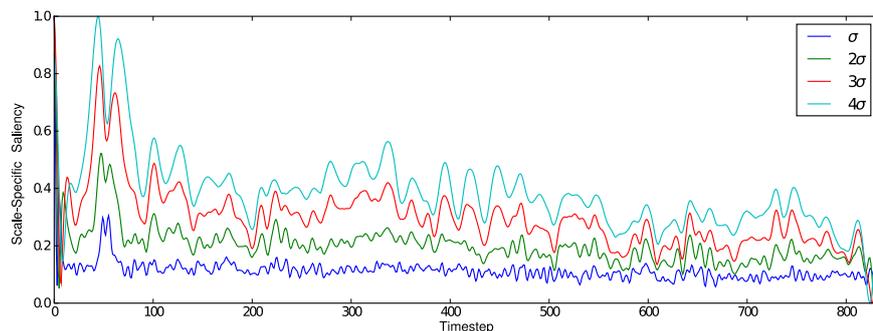
Once we have computed the per-atom saliency for each atom and timestep, we can aggregate information to obtain a per-timestep measure of saliency. We define the saliency of timestep t at scale σ as follows:

$$\mathcal{S}_\sigma[t] = \frac{1}{m} \sum_{a=1}^m \mathcal{S}_\sigma[a, t] \quad (2)$$

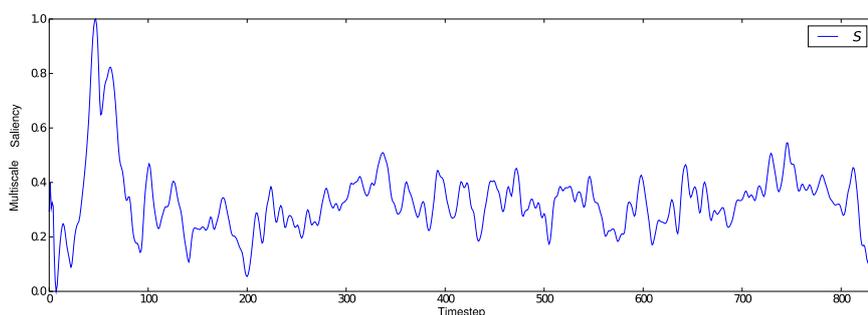
The saliency for timestep t is given simply as the mean saliency of all m constituent atoms at timestep t .

3.1.3 Multiscale Saliency

The definition of saliency we have so far provided exists only at a single scale, σ . Formulating multiscale saliency is as simple as considering multiple values of σ and composing the results.



(a) Saliency Function at Different Scales



(b) Multiscale Saliency Function

■ **Figure 2** Each scale considered produces a different saliency function, as seen in (a). These saliency functions are combined into a single multiscale saliency function \mathcal{S} , as shown in (b).

The multiscale saliency measure for a given timestep t is defined over a set of scales as follows:

$$\mathcal{S}[t] = \frac{1}{|\Lambda|} \sum_{\sigma \in \Lambda} \bar{\mathcal{S}}_{\sigma}[t] \quad (3)$$

Where $\bar{\mathcal{S}}_{\sigma}$ is the normalized saliency function at scale σ and $\Lambda = \{\sigma_1, \sigma_2, \dots\}$ is the set of scales from which the multiscale saliency is composed. Throughout this work, we will simply consider $\Lambda = \{\sigma, 2\sigma, 3\sigma, 4\sigma\}$, though many different choices are possible. The composition of the different single scale saliency functions into the single multiscale saliency function is illustrated in figure 2.

4 Using Saliency

4.1 Interpreting The Saliency Function

Having obtained the multiscale saliency function \mathcal{S} , we may now inquire about its intuitive relationship to the original simulation data. What, precisely, does \mathcal{S} measure? We shall answer this question by again appealing to the criteria by which we defined our saliency function. The computation of \mathcal{S} relies on a center-surround mechanism; effectively determining the uniqueness of a particular timestep with respect to its temporal neighborhood. Furthermore, the computation of \mathcal{S} is multiscale; suggesting that it encodes this uniqueness over temporal neighborhoods of various size.

Thus, we suggest that $\mathcal{S}[t]$ provides a measure of the uniqueness of timestep t with respect to neighboring timesteps over multiple scales. Local extrema of \mathcal{S} may now be interpreted to tell us something about the timesteps to which they correspond. In particular, local maxima of \mathcal{S} correspond to timesteps which are very different, or *anomalous*, with respect to their surroundings. Similarly, local minima of \mathcal{S} correspond to timesteps which are very similar to, or *representative* of, their surroundings. Equipped with this intuitive interpretation of \mathcal{S} , we can now suggest how the saliency function can be used, and how this intuition can be verified.

4.2 Representative Keyframe Selection

As mentioned in section 2, the changes in atomic position between consecutive timesteps of a simulation are dominated by Brownian motion. Only over larger timescales do interesting and purposeful molecular conformational changes occur. This suggests that one might be able to summarize the content of the simulation very precisely by choosing only a few timesteps (keyframes) to represent the entire simulation. Such keyframes can be useful for summarization, indexing, and numerous other tasks.

When finding such keyframes, there is a particular constraint that we wish to respect. Namely, *keyframes should be drawn from the actual simulation data*. Though it might be tempting to produce representative molecular conformations by aggregating information from multiple timesteps of simulation data, there are compelling reasons to avoid this approach. Molecular dynamics simulations are highly computationally intensive and modeled upon equations which respect numerous physical constraints (e.g. atom positions may not superpose, and a molecular configuration should be consistent with the potential field induced by the molecule's constituent atoms). Data representations which aggregate data and synthesize a representation may easily violate such constraints which are painstakingly considered during the course of the simulation. In order to avoid such problems, we shall not consider synthesizing representative keyframes. Rather, the representative keyframes we select will be taken directly from the original simulation data; thus ensuring that they are valid and consistent with the underlying physical model of the simulation.

Respecting this constraint, and understanding its motivation, we may now consider some different approaches to keyframe selection. In particular, we will suggest three methods for keyframe selection.

4.2.1 Random Keyframe Selection

Perhaps the simplest approach to obtaining keyframes is to sample uniformly at random from the full simulation. We shall label this approach as **RS**. Since differences in molecular configuration between nearby timesteps are dominated by Brownian motion, the selection of timesteps which are well distributed and reasonably temporally separated will likely provide a meaningful summary of the simulation. This approach has no dependence on the underlying data, and may yield an arbitrarily poor set of representative keyframes. However, its implementation is completely trivial and exceedingly fast.

4.2.2 Douglas-Peucker Keyframe Selection

The second method we shall consider for keyframe selection is an extension of the classical Douglas-Peucker (**DP**) algorithm [4]. Classically, this algorithm has been used to approximate planar curves by polylines. The algorithm itself is fairly simple and has an elegant recursive

definition. Consider two points, p_1 and p_2 residing on the planar curve \mathcal{C} . Consider the line segment $\overline{p_1 p_2}$ as the linear approximation of \mathcal{C} between these two points. The Douglas-Peucker algorithm considers the orthogonal projection of \mathcal{C} onto $\overline{p_1 p_2}$ at n uniform discrete sample positions between p_1 and p_2 . Let $c^* \in \mathcal{C}$ denote the point for which this orthogonal projection is farthest from the true curve. The **DP** algorithm will add c^* to the set of approximation samples and then recursively descend onto the line segments $\overline{p_1 c^*}$ and $\overline{c^* p_2}$. Various criteria may be established for the termination of the algorithm; for example, one may terminate the algorithm when a desired number of approximation samples has been chosen, or when a maximum acceptable per-sample error threshold has been achieved.

The extension of the Douglas-Peucker algorithm to molecular dynamics simulation data is fairly straightforward. Instead of a planar curve, one attempts to approximate atom positions. Linear interpolation of atom positions is considered between consecutive keyframes, where the temporal distance between these keyframes is used as the linear interpolant. Thus, one may obtain a linear prediction $\vec{p}_{a,t}$ for each atom (a) at each timestep (t) of the true simulation data $\vec{p}_{a,t}$. The difference between the linear prediction of a timestep t and the true simulation data at t can simply be computed as the sum of differences between predicted and actual atom positions as in equation 4:

$$\epsilon_t = \sum_{a=1}^m \left| \left| \vec{p}_{a,t} - \vec{p}_{a,t} \right| \right| \quad (4)$$

In this adaptation of the **DP** algorithm, ϵ_t simply replaces the orthogonal distance between the curve and the approximating line segment; leaving the remainder of the algorithm largely the same.

4.2.3 Saliency Guided Keyframe Selection

Finally, we may consider using the multiscale saliency function \mathcal{S} to guide the selection of keyframes. We shall refer to this approach as saliency guided (**SG**) keyframe selection. Though the intuition behind this method is simple, some care must be taken when actually using \mathcal{S} for keyframe selection. First we must determine precisely how \mathcal{S} should guide the selection of keyframes. In section 4.1, we suggested that local minima of \mathcal{S} correspond to representative timesteps from the simulation. Therefore, it makes sense to choose the timesteps corresponding to the local minima of \mathcal{S} as keyframes. First, however, we must more carefully define what we mean by *local* minima. Since \mathcal{S} is obtained from the composition of saliency functions spanning multiple different scales, it is possible that the fine scale saliency functions will lead to small fluctuations between consecutive values of \mathcal{S} . However, unless the surrounding values of \mathcal{S} are relatively smooth, such small fluctuations should not trigger keyframe selection.

To address this issue, we make use of non-maximal suppression. Given a window size, w , non-maximal suppression will suppress the entire signal with the exception of the w -local maxima. For example, when we consider $\mathcal{S}[t]$ for some timestep t , non-maximal suppression will suppress $\mathcal{S}[t]$ to 0 unless $\mathcal{S}[t] > \mathcal{S}[t+i]$, $i \in [-\frac{w}{2}, \frac{w}{2}]$. A reasonable value of w will ensure that not too many keyframes are selected, and that small and non-meaningful fluctuations in \mathcal{S} do not trigger keyframe selection. Here it is important to note that we consider non-maximal suppression of the function $1 - \mathcal{S}$, which is equivalent to a non-minimal suppression of \mathcal{S} itself.

However, the use of non-maximal suppression introduces a new consideration. How should one choose a *reasonable* value of the window size w ? Recall that w should be related to the

scale in \mathcal{S} at which we wish to detect keyframes. Rather than choosing a fixed value of w , we introduce a procedure for multiscale keyframe selection (**MSKS**). The user provides the **MSKS** procedure with a target number of keyframes, k , and the procedure will return the smallest number $k' \geq k$ of keyframes obtained using a coarse to fine selection procedure. The **MSKS** is possible in part due to the *subset containment* property of non-maximal suppression. Consider performing non-maximal suppression on $1 - \mathcal{S}$ using a window of size w . We shall denote by $NMS(f, w)$ the non-maximal suppression of the function f using a window of size w . Further, let $NZ(f)$ denote the indices for which the discrete function f takes non-zero values. Then we may denote the set of desired keyframes (w -local maxima) as $F_w = NZ(NMS(1 - \mathcal{S}, w))$. Now, consider a window of size $w' < w$. The subset containment property ensures that $F_w \subseteq F_{w'}$. This means that if $1 - \mathcal{S}[t]$ is a w -local maximum, then it is also a w' -local maximum $\forall w' < w$. Bearing this property in mind, we will now describe the **MSKS** procedure.

Algorithm 1: MSKS.

Input: \mathcal{S}, k
Output: F
 $F = \emptyset;$
 $w = |\mathcal{S}|;$
 $f = 1 - \mathcal{S};$
while $|F| < k$ **do**
 $w = \frac{w}{2};$
 $f' = NMS(f, w);$
 for $i = 0; i < |f'|; i = i + 1$ **do**
 if $f'[i] > 0$ **then**
 $F = F \cup i;$
return $F;$

The **MSKS** procedure iteratively builds up a set of keyframes by running non-maximal suppression on $1 - \mathcal{S}$ with windows of decreasing width. The pseudo-code for this algorithm is given in algorithm 1. One important aspect of this algorithm is that when the user requests k keyframes, the algorithm will actually return $k' \geq k$ keyframes. This is due to a natural notion of scale, both in the saliency function \mathcal{S} and in the multiscale keyframe selection procedure. For each window size used during the non-maximal suppression procedure, there are a number of natural keyframes which will be selected at the corresponding scale. This feature is important, because it allows the algorithm to select representations of the underlying simulation at varying levels of coarseness.

Figure 3 shows a plot of the multiscale saliency function \mathcal{S} , where 12 keyframes have been chosen using the **MSKS** procedure. For each keyframe that has been selected, a circle has been plotted at the corresponding local minima of \mathcal{S} . The color of each circle denotes the scale (window size = w), at which the timestep corresponding to that circle was first selected as a keyframe.

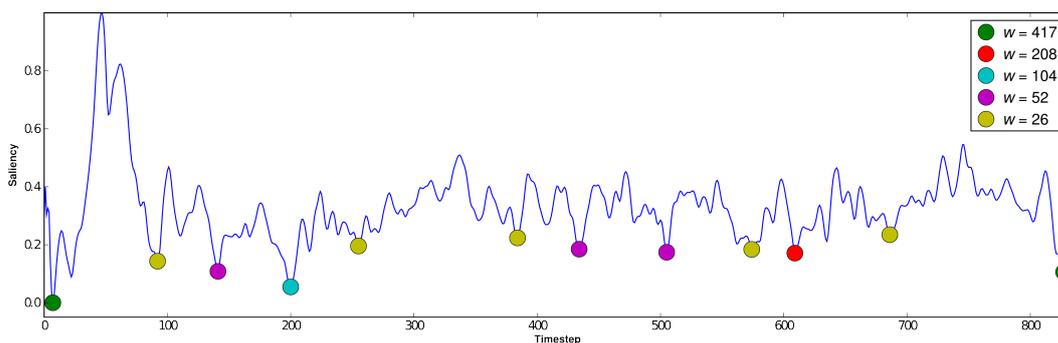


Figure 3 This figure illustrates the selection of keyframes from the multiscale saliency function, \mathcal{S} , using the MSKS procedure. For each timestep chosen as a keyframe, the corresponding local minima is marked with a circle. The color of each circle denotes the non-maximal suppression window size, w , at which the corresponding timestep was first chosen as a keyframe.

5 Results

5.1 Datasets

GroEL Transitions

GroEL is a molecular nanomachine that changes its conformation empowered by ATP chemistry. Spectacular conformational changes between the conformational states of GroEL, such as $T \rightarrow R$ and $R \rightarrow R''$ transitions, are induced upon ATP binding and hydrolysis at the catalytic site of equatorial domain, respectively. Normal operation of the GroEL-GroES chaperonin system [13] is of utmost importance for the cell function as this increases the yield of substrate proteins that are prone to misfold. The misfolding of proteins and subsequent aggregations often lead to the fatal neurodegenerative disorders such as Alzheimers and prion diseases. To better understand the allosteric transitions of GroEL molecule at the microscopic level, multiple sets of Brownian dynamics simulation were performed using a self-organized polymer (SOP) model [7, 5]. The SOP model adopts a strategy of using the minimal representation of proteins and RNA that retains the topological information. The simulations [6] show that $T \rightarrow R$ transition of GroEL, in which the apical domains undergo counterclockwise motion, is mediated by a multiple salt-bridge switch mechanism at the interfaces of seven subunits. The initial event in the $R \rightarrow R''$ transition, during which GroEL rotates clockwise, involves a dramatic outside-in concerted movement of helices K and L , exerting a substantial strain on the GroEL structure, induces the 90 degree clockwise rotation and 40 degree upward movement of apical domain. This simulation consists of 3668 atoms and 834 timesteps.

Folding of Tetrahymena Ribozyme

Large RNAs fold into complex structures which determine their biological activities. The RNA folding problem studies how RNA folds into a unique structure without searching through all possible conformation. How macromolecules from thermophilic organisms achieve thermostability has been a fascinating question for structural biologist and the biotechnology industry. There are many pathways in the folding procedure. Some pathways lead directly to the native state, while others result in “kinetically trapped” conformations that contain some native, as well as non-native interactions. This dataset presents the folding simulation of the

Tetrahymena Ribozyme. Force-quench refolding of the P4-P6 subdomain of the Tetrahymena ribozyme occurs through a compact intermediate. Subsequent formation of tertiary contacts between helices P5b-P6a and P5a/P5c-P4 leads to the native state. This simulation consists of 158 atoms and 1540 timesteps.

5.2 Verification Procedure

It is difficult to visually verify the representative power of the chosen keyframes. Thus, we will rely on a purely quantitative verification procedure. Representative keyframes should be able to predict their temporal neighborhood well. To measure this predictive power, we shall make use of the keyframes chosen by the various selection algorithms (i.e. **RS**, **DP**, and **SG**) and perform an interpolation procedure over them. The total error between the simulation data predicted solely by the keyframes and the actual simulation data will be used as a measure of the predictive/representative power of the selected keyframes. More precisely, for every timestep t which is not, itself, a keyframe, the interpolation procedure will yield a prediction with some resultant error ϵ_t . The representative power of the selected keyframes will be measured by $\epsilon = \sum_{t=1}^n \epsilon_t$, where a smaller ϵ is indicative of more representative keyframes. $\epsilon_{\mathbf{RS}}$, $\epsilon_{\mathbf{DP}}$, and $\epsilon_{\mathbf{SG}}$ are used to indicate the errors using the three methods.

5.3 Experimental Results

The saliency guided keyframe selection approach (**SG**), in paper is compared against the Douglas-Peucker(**DP**) and random selection methods (**RS**); all of which are described in section 4.2. The experiments show the keyframes selected by the saliency approach better approximates the simulations than the other two methods. For each simulation, we consider representing the simulation using three different numbers of key frames, selected using the **SG**, **DP**, and **RS** methods. The random sampling method illustrates the difficulty of this problem and establishes a baseline performance for comparison, and the **DP** algorithm is considered as the current state of the art. Random selection of keyframes are sampled 1000 times for each experiment. The reported performance of the random sampling method is an average across all the samples. The statistics intend to show the error of approximations produced by the two keyframes selection methods (**DP** and **SG**) are lower than random selection, and they are statically significant. The performance difference in between the average random selection and the other two method shows it is unlikely to randomly pick good keyframes to approximate the simulations. This shows that not all timesteps are equally representative, and that it is important to select the correct timesteps when summarizing these complex molecular dynamics simulations.

Table 1 and Table 2 show statistics of the relative improvement between **SG** and **RS**. The relative improvement is computed by:

$$\frac{\epsilon_{\mathbf{RS}} - \epsilon_{\mathbf{SG}}}{\epsilon_{\mathbf{RS}}}$$

Most of the results show the **SG** method performance is 10% - 20% better than the random selection performance.

The null hypothesis here is that the distribution of error due to **SG** is drawn from the same distribution of **RS**. The t -values shown in Table 1 and Table 2 shows this hypothesis can be safely rejected (95% confidence interval).

The relative cumulative approximation error of the experiments with respect to the **RS** method are shown in Figure 4 and Figure 5. The horizontal axis represents the timesteps and

■ **Table 1** Relative improvement statistics of saliency guided (**SG**) frame selection over random selection (**RS**) for the GroEL simulation.

| # keyframes | Relative improvement | stdev | <i>t</i> value |
|-------------|----------------------|--------|----------------|
| 2 | +1.49% | 19.46% | 2.58 |
| 5 | +13.63% | 15.63% | 28 |
| 10 | +18.0% | 8.97% | 63 |

■ **Table 2** Relative improvement statistics of saliency guided (**SG**) frame selection over random selection (**RS**) for the Tetrahymena Ribozyme simulation.

| # keyframes | Relative improvement | stdev | <i>t</i> value |
|-------------|----------------------|-------|----------------|
| 15 | +22.58% | 9.51% | 75 |
| 24 | +22.96% | 6.97% | 94 |
| 45 | +15.44% | 4.2% | 113 |

■ **Table 3** Relative improvement statistics of saliency guided selection (**SG**) over Douglas-Peucker selection (**DP**).

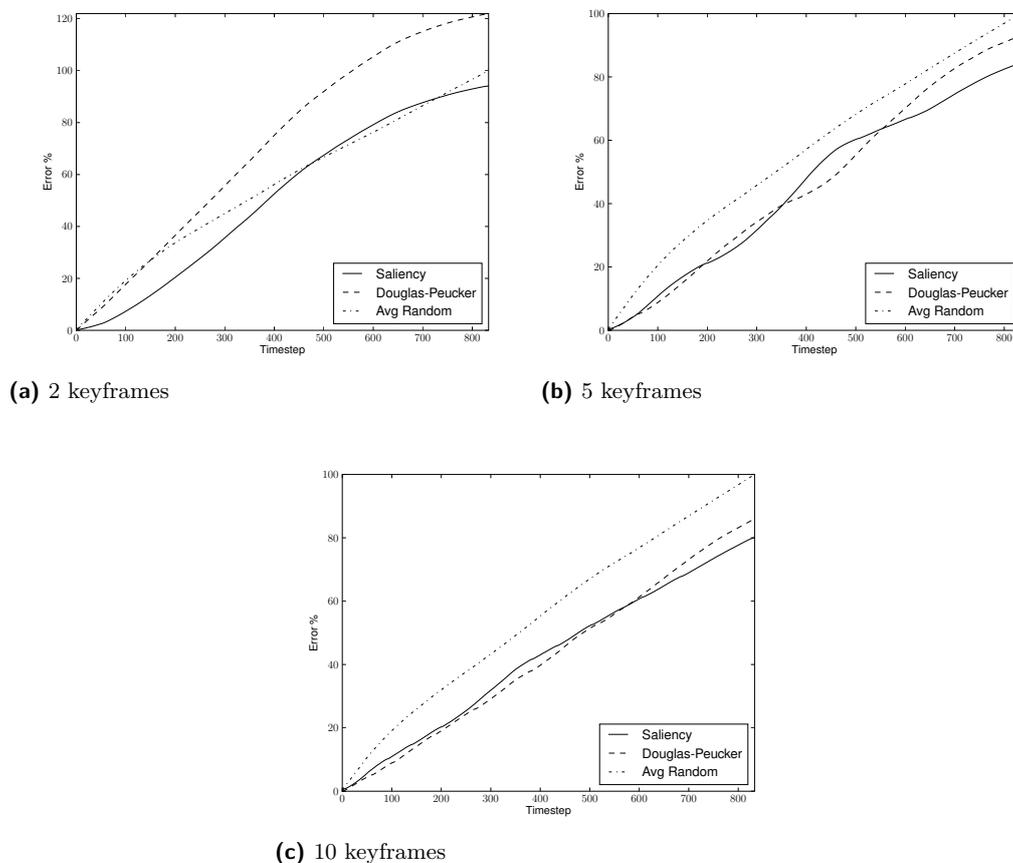
| (a) GroEL | | (b) Tetrahymena Ribozyme | |
|-------------|----------------------|--------------------------|----------------------|
| # keyframes | Relative improvement | # keyframes | Relative improvement |
| 2 | +22.81% | 15 | +12.13% |
| 5 | +9.25% | 24 | +12.12% |
| 10 | +6.61% | 42 | +8.99% |

the vertical axis represents the relative cumulative error. The three plotted lines represent the errors obtained by the **SG**, **DP**, and **RS** methods. Each line shows the cumulative error relative to the **RS** method at a certain timestep throughout the simulations.

In the GroEL simulation, the **SG** method consistently results in the lowest overall error; whereas the 5 or 10 keyframes selected by the **DP** algorithm approximate the simulation better than random selection. The approximations produced by the **SG** method are 6%–22% better than those produced by the **DP** algorithm. In the 2 keyframe experiment, The **DP** algorithm selects only the beginning and the ending timesteps, this results in significant loss of details during the simulation and hence it was even outperformed by average random selection.

In the Tetrahymena Ribozyme simulation, the **SG** method consistently results in the lowest overall error, and the **DP** algorithm also consistently approximates the simulation better than random selection. The approximations produced by the **SG** method are 12%–15% better than those produced by the **DP** algorithm. The trend of the error plot shows that the **SG** method outperformed the **DP** algorithm and **RS** approach at every frame across the simulation.

Table 3 shows a summary of relative improvement obtained by employing the **SG** frame selection method rather than the **DP** frame selection method.



■ **Figure 4** Relative cumulative error (with respect to **RS**) of approximating the GroEL simulation by the selected keyframes.

5.3.1 Selected Keyframes

The selected keyframes from the two simulation are presented in this section.

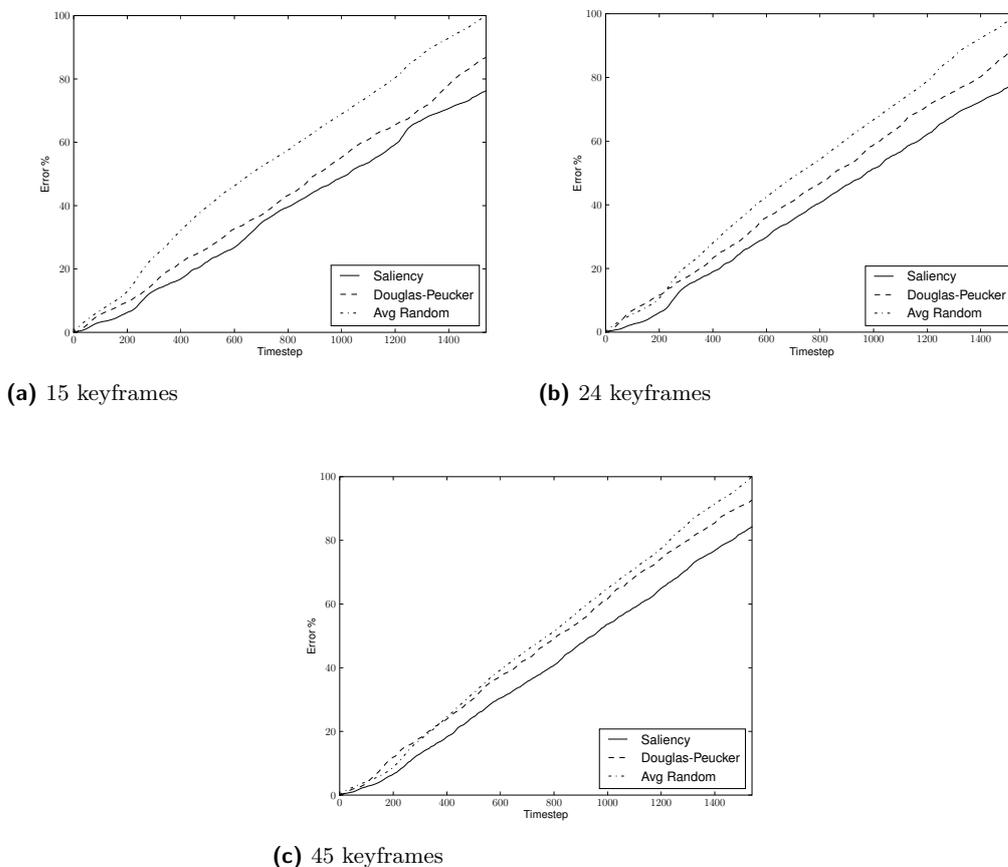
The frames from the GroEL simulation show the conformational changes of the functional subunits during the course of the $R \rightarrow R''$ transition. Figure 6 shows the 10 keyframes selected using the **SG** method for summarizing the GroEL simulation.

The frames from the Tetrahymena Ribozyme simulation show how the Ribozyme molecule folds from a straight chain in to its native state. Figure 7 shows the 10 keyframes selected via the **SG** method for summarizing the Tetrahymena Ribozyme simulation.

5.3.2 The Most Salient Frames

Recall that when using the **SG** method to obtain keyframes, we are interested in obtaining the most representative timesteps from the simulation data. These timesteps are chosen by using the **MSKS** procedure to find the local maxima of $1 - \mathcal{S}$ over various scales.

This procedure naturally leads one to wonder about the effect of using the **MSKS** procedure on \mathcal{S} rather than $1 - \mathcal{S}$. While the local maxima of $1 - \mathcal{S}$ signify the timesteps which are the most similar with respect to their temporal neighborhood over various scales,

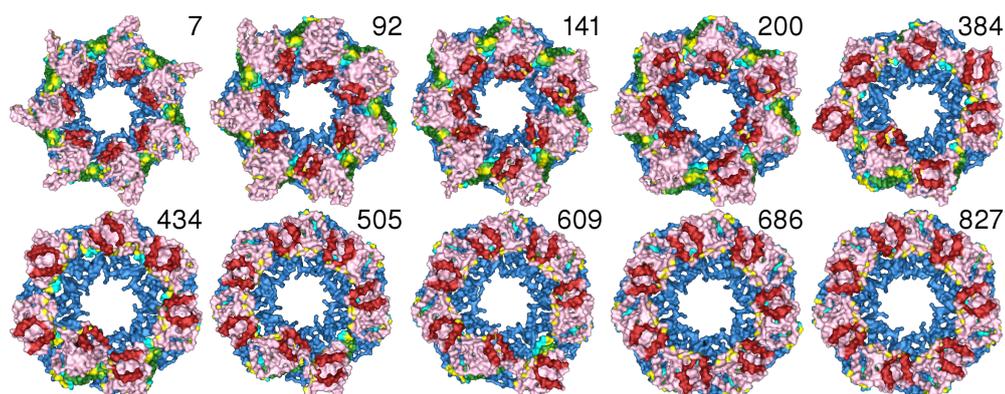


■ **Figure 5** Relative cumulative error (with respect to **RS**) of approximating the Tetrahymena Ribozyme simulation by the selected keyframes.

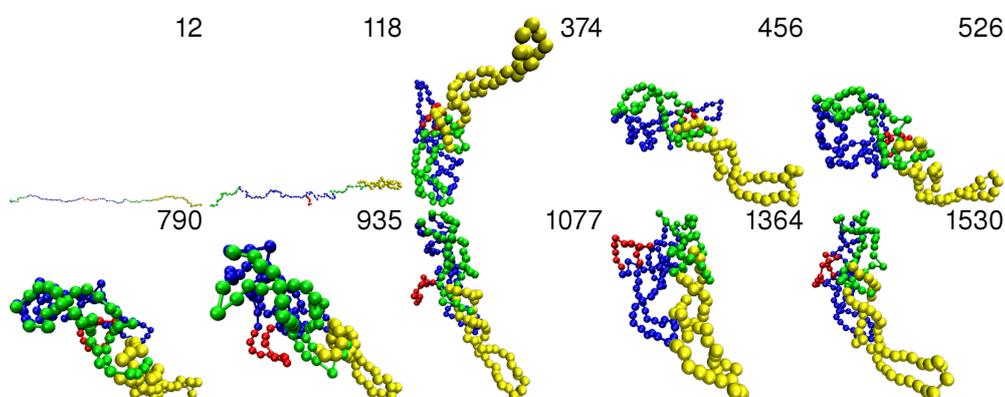
the local maxima of \mathcal{S} correspond to those which are most different. Such anomalous timesteps are useful and informative in their own right. By simply inverting our multiscale saliency function, we are able to select *both the most representative and the most anomalous* timesteps of a simulation. Figure 8 and Figure 9 show the 5 most salient frames from the GroEL and Tetrahymena Ribozyme simulations.

6 Conclusion

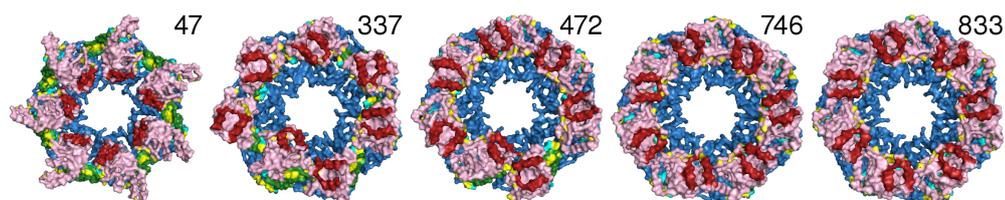
In this work, we have introduced a notion of temporal saliency for molecular dynamics simulation. Such a notion is useful for summarizing, abstracting, indexing, previewing, and analyzing these large time-varying datasets. We have shown how our multiscale saliency function \mathcal{S} can be used in conjunction with a multiscale keyframe selection procedure to choose representative frames from among the locally chaotic motion of molecular dynamics simulations. By employing interpolation over such keyframes as a measure of their predictive power, we have shown that saliency guided keyframe selection consistently chooses more representative frames than other methods.



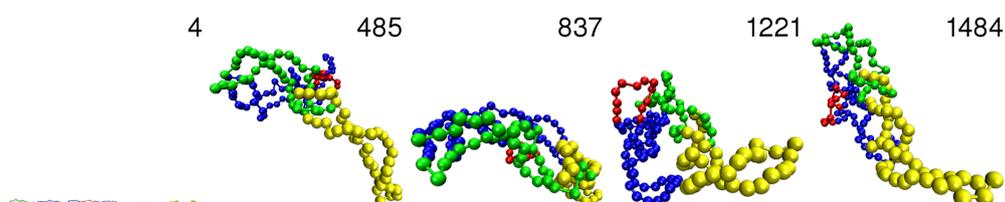
■ **Figure 6** The 10 most representative timesteps from the simulation of GroEL nanomachine consisting of 3668 atoms and 834 timesteps. The corresponding saliency function is shown in Figure 2.



■ **Figure 7** The 10 most representative timesteps from Tetrahymena Ribozyme simulation consisting of 158 atoms and 1540 timesteps.



■ **Figure 8** The 5 most salient timesteps from the 834 timesteps of the GroEL simulation with 3668 atoms.



■ **Figure 9** The 5 most salient timesteps from the 1540 timesteps of the Tetrahymena Ribozyme simulation with 158 atoms.

Acknowledgments

We gratefully acknowledge Andriy Anishkin, Sam Cho, Changbong Hyeon, Sergei Sukharev and D. Thirumalai for providing several datasets as well as expert insight into the corresponding simulations. We also want to thank Youngmin Kim and Dianne P. O’Leary for valuable discussions.

This work has been supported in part by the NSF grants: IIS 04-14699, CCF 04-29753, CNS 04-03313, CCF 05-41120, CMMI 08-35572 and the NVIDIA CUDA Center of Excellence. Any opinions, findings, conclusions, or recommendations expressed in this article are those of the authors and do not necessarily reflect the views of the research sponsors.

References

- 1 Okan Arikan. Compression of motion capture databases. *ACM Transactions on Graphics*, 25(3):890 – 897, July 2006.
- 2 J. Assa, Y. Caspi, and D. Cohen-Or. Action synopsis: pose selection and illustration. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH)*, 24(3):667–676, 2005.
- 3 Daniel DeMenthon, Vikrant Kobla, and David Doermann. Video summarization by curve simplification. In *MULTIMEDIA ’98: Proceedings of the sixth ACM international conference on Multimedia*, pages 211–218, New York, NY, USA, 1998. ACM.
- 4 David H. Douglas and Thomas K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122, 1973.
- 5 C. Hyeon, R. I Dima, and D. Thirumalai. Pathways and kinetic barriers in mechanical unfolding and refolding of RNA and proteins. *eprint arXiv:q-bio/0611050*, November 2006.
- 6 Changbong Hyeon, George H. Lorimer, and D. Thirumalai. Dynamics of allosteric transitions in groel. *Proc. Natl. Acad. Sci.*, 103:18939, 2006.
- 7 Changbong Hyeon and D. Thirumalai. Mechanical unfolding of rna: From hairpins to structures with internal multiloops. *Biophysical Journal*, 92:731, 2007.
- 8 L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. on Pattern Analysis and Machine intelligence*, 20(11):1254–1259, 1998.
- 9 John L. Klepeis, Kresten Lindorff-Larsen, Ron O. Dror, and David E. Shaw. Long-timescale molecular dynamics simulations of protein structure and function. *Current Opinion in Structural Biology*, 19(2):120 – 127, April 2009.
- 10 C. H. Lee, A. Varshney, and D. Jacobs. Mesh saliency. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH)*, 24(3):659 – 666, 2005.
- 11 Mirko Sattler, Ralf Sarlette, and Reinhard Klein. Simple and efficient compression of animation sequences. In *Symposium on Computer Animation*, pages 209 – 217, 2005.
- 12 Deborah Silver and X. Wang. Volume tracking. In *VIS ’96: Proceedings of the 7th conference on Visualization ’96*, pages 157–ff., Los Alamitos, CA, USA, 1996. IEEE Computer Society Press.
- 13 Lorimer GH Thirumalai D. “chaperonin-mediated protein folding”. *Annual Review of Biophysics and Biomolecular Structure*, 30:245–269, 2001.
- 14 Ba Tu Truong and Svetha Venkatesh. Video abstraction: A systematic review and classification. *ACM Trans. Multimedia Comput. Commun. Appl.*, 3(1):3, 2007.