On Representing Protein Folding Patterns Using Non-Linear Parametric Curves

Parthan Kasarapu, Maria Garcia de la Banda, and Arun S. Konagurthu

Abstract—Proteins fold into complex three-dimensional shapes. Simplified representations of their shapes are central to rationalise, compare, classify, and interpret protein structures. Traditional methods to abstract protein folding patterns rely on representing their standard secondary structural elements (helices and strands of sheet) using line segments. This results in ignoring a significant proportion of structural information. The motivation of this research is to derive mathematically rigorous and biologically meaningful abstractions of protein folding patterns that maximize the economy of structural description and minimize the loss of structural information. We report on a novel method to describe a protein as a non-overlapping set of parametric three dimensional curves of varying length and complexity. Our approach to this problem is supported by information theory and uses the statistical framework of minimum message length (MML) inference. We demonstrate the effectiveness of our non-linear abstraction to support efficient and effective comparison of protein folding patterns on a large scale.

Index Terms—Protein folding patterns, protein abstractions, minimum message length

1 Introduction

Globular proteins fold into complex three-dimensional (3D) shapes [1]. The essence of their folding patterns is captured by the geometry of their secondary structural elements—helices and strands of sheet [2], [3].

Simplified representations of folding patterns, like the ones at the level of secondary structures, are tremendously useful to understand the architecture and topology of protein structures. With the rapidly growing world wide protein data bank (wwPDB), simplified representations allow efficient and effective methods for large-scale search, alignment and classification handling the whole corpus of structures.

Almost universally, current methods to represent protein folding patterns rely on secondary structural elements, which are then replaced by line segments or vectors summarizing them [2], [4], [5]. This has resulted in very useful and compact summaries of protein folding patterns.

Although abstractions at the level of secondary structural elements are very compact, representations at this level are inconsistent and lossy in terms of the structural information they capture. On average, standard secondary structures account for 60-70 percent of globular protein chains, leaving the structural information in the remaining 30-40 percent of the chain entirely ignored [6]. Also, due to the lack of consensus in secondary structure assignment, such abstractions tend to be inconsistent [6].

A few methods have been proposed to abstract protein folding patterns that capture their essence without discarding structural information [7], [8]. Mainly, these methods summarize a given protein backbone using *piecewise* line segments. These representations are *independent* of the notion of secondary structures and hence, allow for the consideration of regions that do not belong to the conventional classes of helices and strands [7], the information that is lost in abstractions at the secondary structure level.

However, the main disadvantage of using piecewise linear abstractions relates to the fact that structures are flexible and undergo plastic deformations in protein evolution. Representing the protein backbone using rigid lines does not accommodate for the flexibility and plasticity required to describe protein folding patterns concisely. Our work addresses and rectifies this limitation.

The primary goal of any abstraction should be to maximize the economy of description of a protein structure while ensuring that the essence of its folding pattern is retained [7], [8]. We propose here a mathematically rigorous approach to represent any protein structure using non-linear parametric curves. Our approach relies on the rigorous information theoretic criterion of minimum message length (MML) inference.

Here, we investigate the compressibility of protein coordinate data using non-linear parametric curves. While our method is applicable to any class of parametric curves, this work specifically employs Bézier curves for their robustness in modelling regions of protein structures effectively. Bézier curves are smooth and continuous, and easily adapt to the plasticity commonly observed in proteins. These curves can be scaled to any size and are simply defined by their *control points*. The number of control points determines the *degree* (or order) of the curve. A linear Bézier curve has two control points and is of degree 1, a quadratic Bézier curve has three control points and is of degree 2, a cubic Bézier curve is of degree 3 containing four control points and so on. In general, the first and last control points of Bézier curves always lie on their respective curve. The intermediate control points (where they exist) lie away from the curve. The freedom to move the intermediate control points give Bézier curves their flexibility and therefore, they are ideal to describe the observed plasticity in proteins.

• *The authors are with the Clayton School of Information Technology, Monash University, VIC 3800, Australia. E-mail: {parthan.kasarapu, maria.garciadelabanda, arun.konagurthu}@monash.edu.*

Manuscript received 9 Jan. 2014; revised 28 May 2014; accepted 1 July 2014. Date of publication 10 July 2014; date of current version 4 Dec. 2014.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TCBB.2014.2338319

Our method relies on *dissecting* (or segmenting) any given protein structure into non-overlapping, variable-length regions (or segments), each of which is assigned to a Bézier curve of an arbitrary degree (or complexity). Each segmented region in the dissection has a statement cost corresponding to the amount of information (measurable in *bits*) required to losslessly encode the coordinate information in that region, using its assigned Bézier curve as the model for compression. (Hence, we use the term *code length* to define the information cost associated with each dissected region.)

Furthermore, we design a search strategy to find the *optimal* dissection (and its corresponding Bézier curve assignment) that results in the shortest lossless encoding of protein coordinates using an ensemble of Bézier curves, where each curve in the ensemble defines a particular dissected region in the protein.

The paper is organized as follows. Section 2 introduces the MML inference framework. Section 3 builds the encoding schemes to losslessly compress protein coordinates using Bézier curves. Section 4 presents the details of our strategy to search for the optimal dissection under the MML objective function. Section 5 compares and contrasts the dissection with various other methods of structural representation.

2 Minimum Message Length (MML) Inference

2.1 Introduction

Wallace and Boulton [9] developed the first practical criterion for model selection using information theory. The resulting MML framework provides a statistically rigorous approach to objectively compare any two competing hypotheses and, hence, choose the best one.

Formally, Bayes's theorem gives

$$\Pr(H \,\&\, D) = \Pr(H) \times \Pr(D|H) = \Pr(D) \times \Pr(H|D),$$

where D denotes some observed data, and H some hypothesis on that data. Further, $\Pr(H \,\&\, D)$ is the joint probability of data D and hypothesis H, $\Pr(H)$ is the prior probability of hypothesis H, $\Pr(D)$ is the prior probability of data D, $\Pr(H|D)$ is the posterior probability of H given D, and $\Pr(D|H)$ is the likelihood.

MML uses the following result from Shannon's information theory [10]: given an event or outcome E whose probability is $\Pr(E)$, the length of the optimal lossless code $I(E)$ to represent that event requires $I(E) = -\log_2(\Pr(E))$ bits.

Applying Shannon's insight to Bayes's theorem, Wallace and Boulton [9] got the following relationship between conditional probabilities in terms of optimal message (or code) lengths:

$$I(H \,\&\, D) = I(H) + I(D|H) = I(D) + I(H|D).$$

As a result, given two competing hypotheses H and H', we know that

$$I(H \,\&\, D) - I(H' \,\&\, D) = I(H) + I(D|H) - I(H') - I(D|H').$$

The framework can be best understood as a communication process where an imaginary pair of transmitter and receiver are connected through a Shannon communication channel. The objective is for a transmitter to send the data D across to the receiver. The transmitter and receiver must have previously agreed on a set of rules (that is, a *code book*) of communication, containing no more than what is common knowledge and prior beliefs. The transmitter chooses a hypothesis H to fit the data D, and uses H to encode D and transmits D to the receiver *such that the receiver should then be able to reconstruct D losslessly*. To achieve this, the transmitter sends the encoded information in two parts: the hypothesis H (which takes $I(H)$ bits) and the observed data D using the knowledge of H (which takes $I(D|H)$ bits).

Clearly, the message length can vary depending on the complexity of H and how well it can explain D. A more complex H may fit (i.e., explain) D better but takes more number of bits to state itself. The trade-off comes from the fact that the transmission process requires the encoding of both the hypothesis and the data given the hypothesis, that is, the model complexity $I(H)$ and the goodness of fit $I(D|H)$.

2.2 Formulating the Problem Using MML

The data in our problem are the three dimensional coordinates of a given protein $\mathcal{P}$. This protein will be represented here as the sequence of n 3D points $\{P_1, \ldots, P_n\}$, representing the coordinates of the C_α atoms along the N- to C-terminus of the protein chain.

A non-linear abstraction of $\mathcal{P}$ defines a subsequence containing $k < n$ points from $\mathcal{P}$ denoted in this work as $\mathcal{Q} = \{Q_1 \equiv P_{i_1}, Q_2 \equiv P_{i_2} \ldots, Q_k \equiv P_{i_k}\}$ s.t. $1 = i_1 < i_2 < \cdots < i_k = n$. Any *successive* pair of points $(Q_r, Q_{r+1})_{1 \le r < k} \in \mathcal{Q}$ defines a contiguous region in the protein structure whose end points are $Q_r = P_{i_r}$ and $Q_{r+1} = P_{i_{r+1}}$. We use the term *dissection* to indicate the collection of regions defined by $\mathcal{Q}$. Associated with each region $Q_r \cdots Q_{r+1}$ in the dissection is a Bézier curve of some degree θ_r with (Q_r, Q_{r+1}) acting as the start and end control points of that curve. The remaining $(\theta_r - 1)$ control points are determined analytically by minimizing the total least squares errors of the set of points in that region with respect to the Bézier curve.

Translating our work using the MML paradigm described in Section 2.1, any dissection $\mathcal{Q}$ (and its corresponding Bézier curve assignment) denotes a hypothesis that attempts to concisely summarize a given protein structure. The best dissection in this framework is the one that gives the shortest encoding (i.e., the best compression) of the entire set of coordinates in $\mathcal{P}$, over all possible hypotheses to describe $\mathcal{P}$.

3 Communicating the Hypothesis and the Data

3.1 Encoding the Hypothesis

The first part of the transmission involves communicating the hypothesis, that is, the dissection $\mathcal{Q}$ containing k segments and its corresponding Bézier curve assignment. This is achieved using the following steps.

1) *Transmit the number of segments.* We use the Elias omega code [11], which is a variable length integer code, to encode k. This takes $\log^* k = \log k \; \log k + \cdots$ (over all positive terms) bits to encode k.
2) *Transmit the end points.* The end point of the previous segment in any dissection is also the start point of the current one. (The first segment is a special case

where the transmission of the start point can be avoided if the coordinates in $\mathcal{P}$ are translated such that P_1 is always the origin.) The coordinates of the end point of each segment are three real numbers of the form (x, y, z). To transmit these coordinates a bounding box is specified using $(x_{\min}, y_{\min}, z_{\min})$ and $(x_{\max}, y_{\max}, z_{\max})$ which can be determined from the set of all coordinates. Each end point can be stated in $\log V$ bits, where $V = (x_{\max} - x_{\min}) \times (y_{\max} - y_{\min}) \times (z_{\max} - z_{\min})$ is the volume of the bounding box. Hence, to encode the k end points requires a statement cost of $k \log V$ bits.

3) *Transmit intermediate control points.* Associated with any segment $Q_r \equiv P_i \cdots Q_{r+1} \equiv P_j$ in dissection $\mathcal{Q}$ is a Bézier curve of degree θ_r, containing $\theta_r - 1$ intermediate control points dictating its curvature. The degree θ_r is encoded using the same aforementioned integer code, taking $\log^* \theta_r$ bits to encode. The spatial positions of the control points are stated using the bounding box approach described above. The length of encoding the $(\theta_r - 1)$ intermediate control points associated with each segment is $(\theta_r - 1) \log V$ bits.

Adding each of the above contributions to the message length required for the first part, we get:

$$I_{\text{part1}} = \log^*(k) + k \log V + \sum_{r=1}^{k} (\log^* \theta_r + (\theta_r - 1) \log V). \quad (1)$$

3.2 Encoding the Data Given the Hypothesis

The second part of the message consists of transmitting the remaining protein coordinates, given the dissection (and assigned Bézier curves) as the hypothesis. This is achieved using the following steps.

1) *Transmit the number of internal points within a region.* For segment r between P_i and P_j, there are $j - i - 1$ *internal* points that need to be encoded using the assigned Bézier curve as their model of compression. Again, the same integer encoding described previously is used taking $\log^*(j - i - 1)$ bits.
2) *Transmit the coordinates of the internal points.* Given that the receiver knows the dissection and its assigned Bézier curves, the internal points corresponding to any segment can be explained as a set of three spatial deviations with respect to its curve. Using these deviations, the receiver can precisely locate (to the stated precision) the position of the internal protein residue. The three sets of spatial deviations are transmitted over a probability distribution, the parameters of which need to be encoded as part of the transmission.

Fig. 1 illustrates the encoding of this part. Let C_{ij} be the curve that abstracts the protein segment between P_i and P_j. We use this curve in conjunction with a plane to explain the internal points that lie between P_i and P_j. To build the plane, we use three non-collinear points P_i, P_j and the first intermediate control point (if it exists) of the assigned Bézier curve. If no such control point exists (i.e., when the assigned Bézier curve is of degree 1), we use P_{i+1} (and transmit it using the bounding box approach).

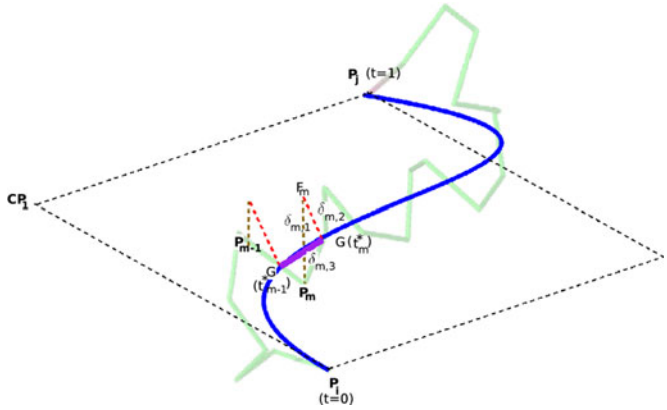

Fig. 1. Deviations of internal points of a region with respect to the assigned (cubic) Bézier curve.

3.2.1 Computation of Deviations

Fig. 1 shows two internal points P_{m-1} and P_m within a segment defined by P_i and P_j as its end points. A plane is defined for this segment using P_i, P_j, and CP_1. Each internal point is associated with three spatial deviations computed as follows:

1) *Deviation 1.* The first deviation is the orthogonal projection of the internal point onto the defined plane. The foot of the perpendicular line from P_m onto the plane is denoted by F_m. The length of this projection is denoted by a signed deviation $\delta_{m,1}$. The sign of this deviation is determined by the orientation of the P_m with respect to the plane (i.e., above or below defined by the normal vector $\hat{n}$ of that plane).
2) *Deviation 2.* The second deviation denotes the shortest distance from the foot of the perpendicular F_m, to the Bézier curve C_{ij} assigned to this region.[1] Let G_m denote the projection on the curve which results in the shortest distance. The length of this projection is given by the signed deviation $\delta_{m,2}$. The sign is determined by the orientation of F_m with respect to the plane formed by the tangent at G_m and the normal $\hat{n}$.
3) *Deviation 3.* Every point on the Bézier curve, including G_m, is parameterized using t in the range $[0, 1]$. Let t^*_m be the parameter of G_m. Similar to the projection G_m of P_m, the previous intermediate point P_{m-1} has an associated projection G_{m-1}. Let t^*_{m-1} be the parameter corresponding to G_{m-1}. The third deviation is the offset of G_m from G_{m-1}, whose sign is given by the position of t^*_m with respect to t^*_{m-1}.

3.2.2 Encoding the Deviations

For the segment r between P_{i_r} and P_{j_r} in the dissection, let the number of points whose deviations needs to be transmitted be $n_r = j_r - i_r$. Each of these n_r points has three spatial deviations. Let δ^r_1, δ^r_2, and δ^r_3 correspond to

1. The details of this computation are described in Appendix A (in the supplemental material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TCBB.2014.2338319).

the set of the first, second, and third spatial deviations respectively. That is, $\delta_p^r = \{\delta^r_{(i+1,p)}, \delta^r_{(i+2,p)}, \ldots, \delta^r_{(n_r-1,p)}\}$ for $p = \{1, 2, 3\}$. To explain these sets of deviations, we consider a statistical distribution. The communication of these sets of deviations over this statistical distribution will require the transmission of the parameters of the distribution, followed by the encoding of the observed deviations using that distribution.

In this work, each set of deviations is encoded using a Normal distribution. We note that the *mean square error* of the deviations with respect to its Bézier curve corresponds to the variance of the estimator. The parameters of the distribution which result in the minimum message length to encode the data are inferred using the Wallace-Freeman approximation [12] and the resultant expression for message length to encode any set of deviations (δ_p^r) is given by the following formula (see [13]):

$$I_{\delta_p^r} = 1 + \log\kappa_2 + \log\left(\mathrm{R}_\mu \mathrm{R}_\sigma\right) + \frac{1}{2}\log\left(2n_r^2\right) + \frac{n_r}{2}\log\left(\frac{2\pi}{\epsilon^2}\right) + \frac{n_r - 1}{2}\log\left(\frac{\sum_{i=1}^{n_r}\left(\delta^r_{(i,p)} - \hat{\mu}_p\right)^2}{n_r - 1}\right) + \frac{n_r - 1}{2},$$

where $\hat{\mu}_p = \frac{1}{n_r}\sum_{i=0}^{n_r}\delta_{(i,p)}$ is the MML estimate of the mean of the distribution, and $\kappa_2, \mathrm{R}_\mu \mathrm{R}_\sigma$ are hyperparameters used in MML inference.

Therefore, the statement cost to encode the coordinates of any segment r between points P_{i_r} and P_{j_r} is given by $I_{\delta^r} = \log^*(n_r) + \sum_{p=1}^{3} I_{\delta_p^r}$.

3.2.3 Total Cost of Communicating the Coordinates Using Bézier Curves

Given the dissection $\mathcal{Q}$ (hypothesis) of the coordinates $\mathcal{P}$ (data), we denote the total message length required to explain the data as $\mathcal{I}(\mathcal{Q}\,\&\,\mathcal{P})$. Combining the code lengths to state the two part message described in Sections 1 and 2 we have that

$$\mathcal{I}(\mathcal{Q}\,\&\,\mathcal{P}) = \log^*(k) + k\log V + \sum_{r=1}^{k}(\log^*\theta_r + (\theta_r - 1)\log V) + \sum_{r=1}^{k} I_{\delta^r}.$$

This equation serves as the objective function that must be minimized in our work. Noting that $\log^*(k)$ is very small and is drastically less than $k\log V$, we ignore this term to construct a slightly modified objective. The advantage of this modified objective is that it is strictly additive in terms of the code lengths corresponding to each individual segment in any dissection of the proteins into Bézier curves. More specifically,

$$\mathcal{I}(\mathcal{Q}\,\&\,\mathcal{P}) \approx \tilde{\mathcal{I}}(\mathcal{Q}\,\&\,\mathcal{P}) = \sum_{r=1}^{k}\mathcal{H}_{ij}^{\theta_r},$$

such that

$$\mathcal{H}_{ij}^{\theta_r} = \theta_r \log V + \log^*(\theta_r) + I_{\delta^r}, \tag{2}$$

where $\mathcal{H}_{ij}^{\theta_r}$ denotes the component code length required to explain the coordinates in the region between P_i^r and P_j^r in the dissection, using a Bézier curve of degree θ_r.

3.2.4 The Optimization Problem

This sets up the search problem of our work of finding the dissection $\mathcal{Q}^*$ of a set of coordinates of protein $\mathcal{P}$ such that the encoding length of $\mathcal{P}$ using $\mathcal{Q}^*$, i.e., $\mathcal{I}(\mathcal{Q}^*\,\&\,\mathcal{P})$, is the *minimum* over the entire space of possible dissections.

4 Finding the Optimal Bézier Curve Dissection

This section will describe the procedure to compute the optimal dissection $\mathcal{Q}^*$ and its associated Bézier curve assignment for the given protein coordinate data $\mathcal{P} = \{P_1, \ldots, P_n\}$. Potentially, every pair of points P_i and P_j, $1 \le i < j \le n$ defines a candidate region that the optimal dissection $\mathcal{Q}^*$ could include. Furthermore, associated with each candidate region is the assignment of a specific Bézier curve of arbitrary degree θ.

In order to search for the best dissection, an *ensemble* of code length matrices $\mathcal{H}^\theta = (\mathcal{H}_{ij}^\theta)_{\forall 1 \le i < j \le n}$, one for each degree $\theta = \{1, 2, 3, \ldots\}$ of the family of possible Bézier curves, is constructed.[2] Each $\mathcal{H}_{ij}^\theta$ contains the code length to encode the coordinates of the region $P_i \ldots P_j$, using a Bézier curve of degree θ. (Section 3 covers the details of computing this code length.)

This ensemble of code length matrices $\mathcal{H}^\theta$ is then used to search for the best dissection $\mathcal{Q}^*$ and its corresponding Bézier curve assignment. As described in Section 3, the goal is to find the dissection that yields a *globally* minimum message length, $\mathcal{I}(\mathcal{Q}^*\,\&\,\mathcal{P})$, to encode the coordinate data. Given the property (see Equation (2)) that the code lengths to encode individual segments of any dissection are strictly additive, the best dissection can, therefore, be derived using a one-dimensional dynamic program with the following recurrence relationship (starting from the boundary value $C_1 = 0$):

$$C_j = \min_{i=1}^{j-1}\left\{\begin{array}{l}\min_\theta \mathcal{H}_{1j}^\theta \\ \left(C_i + \min_\theta \mathcal{H}_{ij}^\theta\right)\end{array}\right., \quad \forall 1 \le j \le n$$

where any C_j gives the optimal dissection from P_1 up to some intermediate point P_j ($1 < j \le n$). We note, that the above recurrence ensures that the optimal dissection of coordinates P_1 to P_j is built incrementally using the optimal dissection of its subproblems defined by $P_1, \ldots, P_i, \forall 1 \le i < j < n$, using the ensemble of code length matrices which are precomputed before the search begins. At the end of the recurrence, the value C_n corresponds to the minimum message length corresponding to the optimal dissection of $P_1, \ldots, P_n$, $I(\mathcal{Q}^*\,\&\,\mathcal{P})$. This dissection and the corresponding Bézier curve assignment (which are *memoized* when computing each C_j) can

2. In practice, we consider only three matrices, by restricting the degree of Bézier curves to *at most* 3. We note that the ensemble of linear, quadratic and cubic Bézier curves are sufficiently powerful to explain the observed plasticity in protein structures.

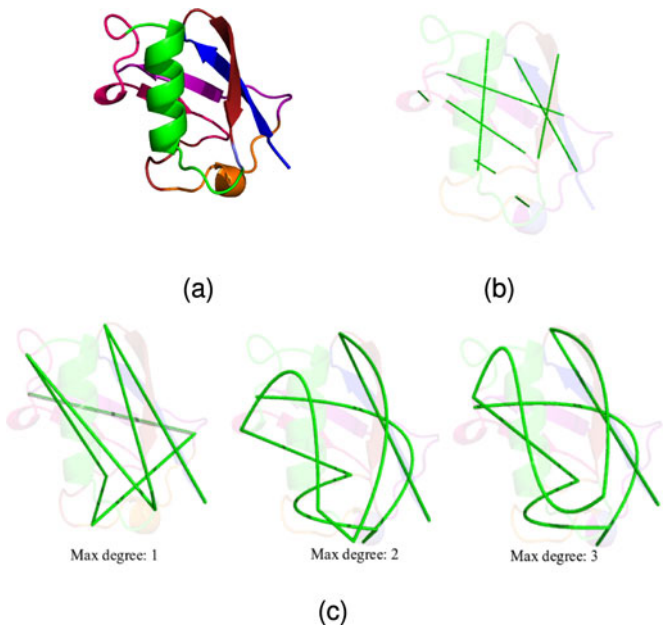

Fig. 2. (a) Structure of Ubiquitin-like domain of human homologue a of RAD23 (wwPDB code 2WYQ) shown in a cartoon representation with standard secondary structures—helices and strands of sheet assigned using DSSP [15]. (b) Traditional representation of the folding pattern which replaces each secondary structural element with line segments. (c) The representations produced by our approach by constraining the Bézier curves to a maximum degree of 1—linear (left frame), maximum degree of 2—linear and Quadratic (middle frame), and maximum degree of 3—linear through to cubic (right frame).

be computed by *backtracking* along the array of dynamic programming history values given by C.

5 Results

5.1 Case Study: Dissection of Regions of Ubiquitin-Like (UBL) Domain of Human Homologue A of RAD23

We illustrate the nature of the dissections generated using our approach, using (randomly chosen) Ubiquitin-like domain of Human homologue A of RAD23 [14] with wwPDB code 2WYQ. We also compare and contrast the representations derived from our approach with the traditional representation at the level of secondary structural elements.

Fig. 2a shows the structure of 2WYQ with standard secondary structures—helices and strands of sheet—represented as cartoons, assigned using DSSP [15]. In the traditional approach to abstracting protein folding patterns, the secondary structural elements are replaced by line segments [16], [17]. This is shown in Fig. 2b. Notice that in excess of 50 percent of the structure is omitted as they do not partake in forming local secondary structural features. As a result, the representation remains lossy.

Fig. 2c shows three dissections of 2WYQ produced by our approach described above, by varying the maximum degree of Bézier curves used to dissect the protein. In the left frame of the figure, we constrain the approach to utilize only the linear order (degree $=1$) Bézier curves. This results in a piecewise linear abstraction of protein folding pattern. This representation is equivalent to those produced by STICKS [7] and PMML [8]. However the strength of the methodology

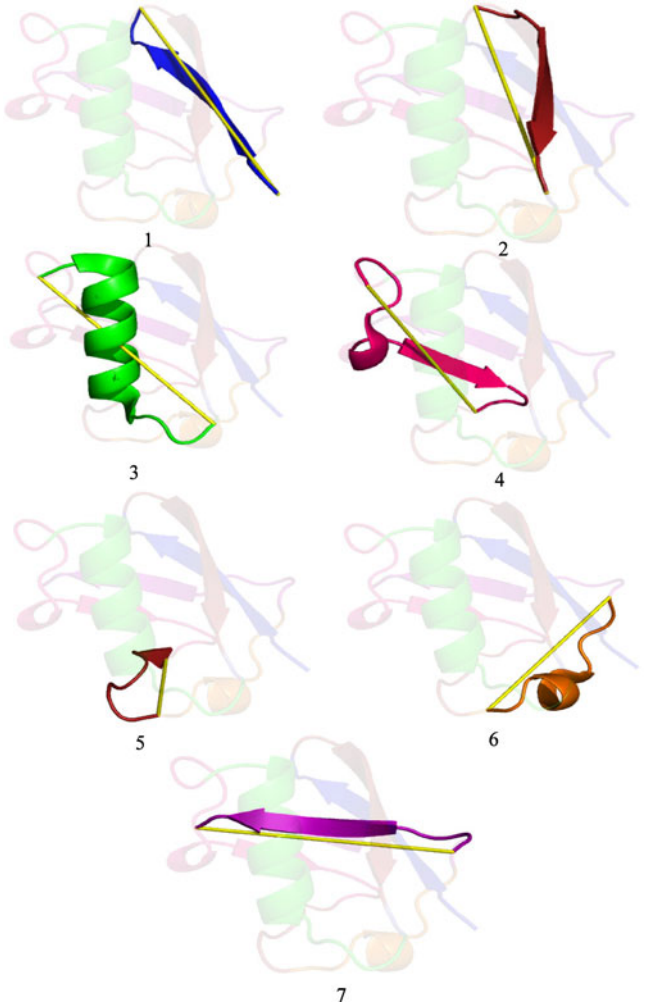

Fig. 3. Linear Bézier curve abstractions of 2WYQ. Note, each highlighted segment is reoriented to the front for clarity of view.

described in this paper is that our method can be generalized to higher order curves, with the piecewise linear approximation being the limiting case. The middle frame of Fig. 2c shows the dissection when the available models include both linear (degree 1) and quadratic (degree 2) Bézier curves. The right frame of the figure further extends this model set to include cubic Bézier curves (degree 3).

5.1.1 Discussion on the Generated Bézier Abstractions

To understand the quality of dissections produced by our approach under varying constraints on the degree of Bézier curves utilized for abstraction, we compare the maximum degree 1 and maximum degree 3 dissections reported in Fig. 2c. In the rest of the text we will refer to these two dissections as *linear* and *non-linear* Bézier abstractions respectively.

The linear Bézier abstraction yields eight segments while the non-linear Bézier abstraction results in seven segments. These segments are individually shown in Figs. 3 and 4. (There are two more segments but they are only two residues long; these will be ignored in our discussion below). Noteworthy aspects of the two abstractions are:

- Segment 1 corresponds to a strand of a four-stranded anti-parallel β-sheet in 2WYQ. This strand shows a mild curvature (see Fig. 3(1)). The non-linear

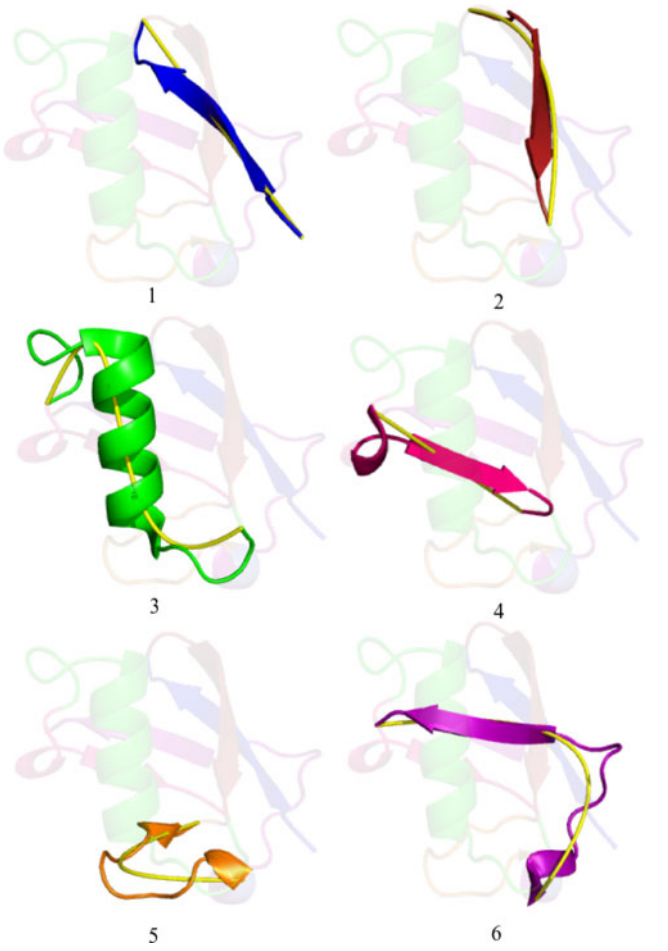

Fig. 4. Non-linear Bézier curve abstractions of 2WYQ. Note, each highlighted segment is reoriented to the front for clarity of view.

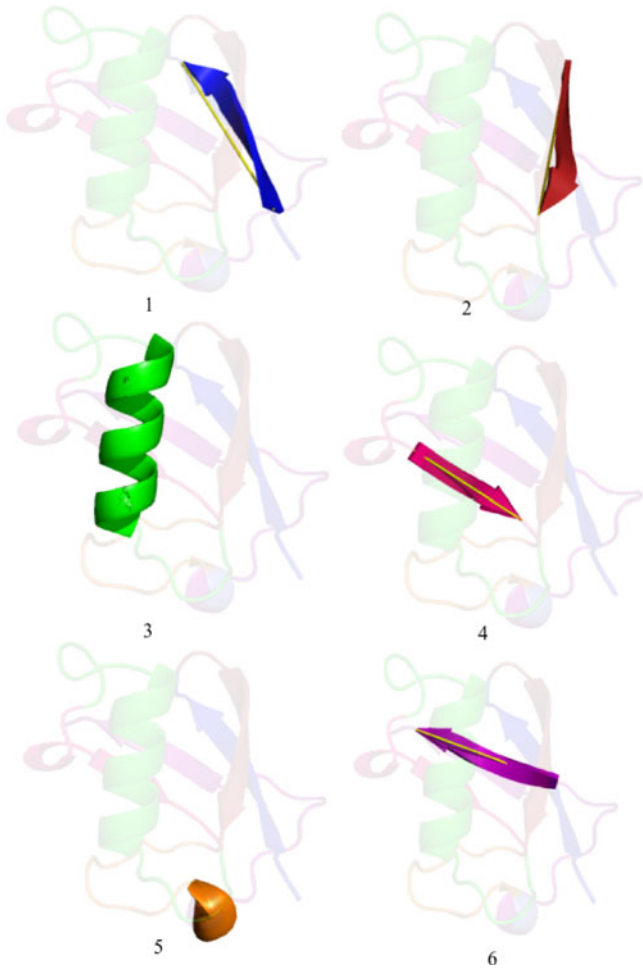

Fig. 5. Segmentation generated using DSSP.

abstraction suitably captures this curvature using a quadratic Bézier curve (Fig. 4).

- Segment 2 corresponds to the anti-parallel strand with respect to the first, and shows a moderate curvature. The linear Bézier abstraction introduces a poorly fitting line segment, while the non-linear Bézier abstraction chooses a well fitting quadratic Bézier curve to explain that region. (see Fig. 4(2)).
- Segment 3 comprises mainly of a helical regions whose terminal regions are flanked by loops. Demonstrating the economy of abstraction, the non-linear Bézier abstraction models this region using a cubic curve (see Fig. 4(3)). In stark contrast, the linear Bézier abstraction models the region using a poorly fitting line segment (see Fig. 3(3)), thereby losing the information of the curvature in that region.
- Segment 4 has a linear trend and hence both dissections model this region using a line segment (see Figs. 3(4) and 4(4)).
- Segment 5 is a coil and this is approximated using a quadratic Bézier curve in the non-linear abstraction (see Fig. 4(5)). The linear abstraction, however, models this region partially using two line segments (see Fig. 3).
- Segment 6 in the non-linear abstraction (see Fig. 4(6)) comprises of a coil and a β strand. This combination is described in the non-linear abstraction using a single quadratic Bézier curve. The linear abstraction for this segment results in two segments—a part of the coil belongs to segments 6 and the beta sheet is represented as a line (Figs. 3(6) and 3(7)).

5.1.2 *Comparison with Secondary Structural Segmentations*

We further compare our non-linear Bézier abstraction with the representation of folding patterns using secondary structures. We note that several comparative studies have highlighted the poor consensus among popular secondary structural assignment programs [6]. Therefore, we consider two radically different programs to assign secondary structures to protein coordinate data: DSSP [15] and SST [18]. The coordinates of 2WYQ were passed through DSSP and SST and the resulting segmentations are used to represent the backbone of this structure as a set of vectors (by replacing the secondary structural elements with line segments; see Figs. 5 and 6).

Using both these methods a major portion of the protein folding pattern goes unrepresented (out of 77 residues, the segmentations generated using DSSP and SST result in representing 37 and 44 residues respectively).

Visual inspection of the Fig. 2a reveals that the secondary structural elements in 2WYQ depart considerably from their ideal geometries and hence the line segments

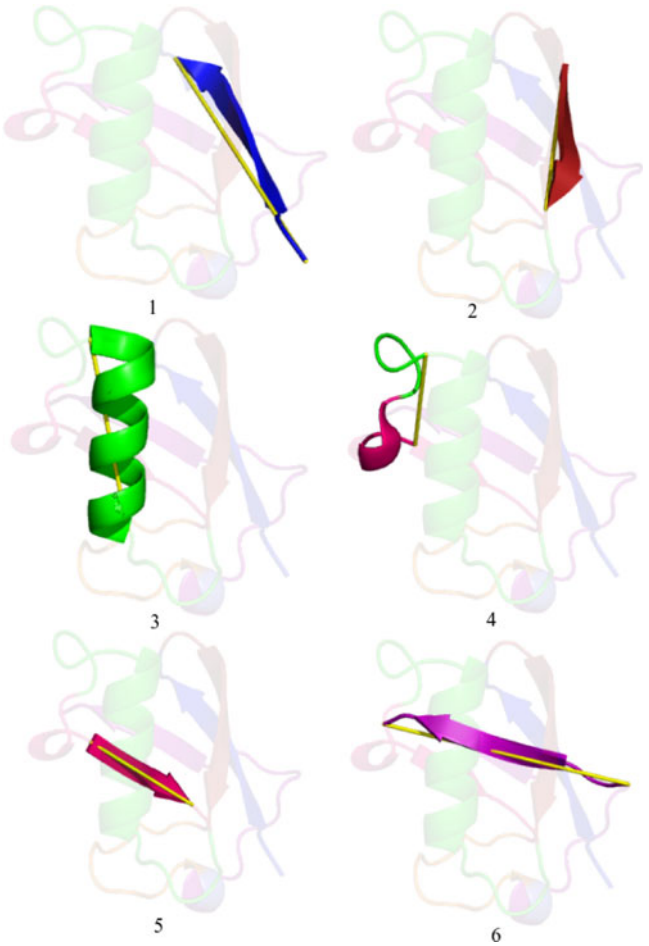

Fig. 6. Segmentation generated using SST.

approximating these regions, for both the approaches, show significant errors. In contrast, our non-linear abstraction results in a well defined abstraction that closely approximates the folding pattern of 2WYQ. This case study and many others we have carefully examined validates the effectiveness of the non-linear Bézier abstractions produced by our approach. It uses an elegant and mathematically rigorous approach to achieve the objective of maximizing the economy of description, while at the same time minimizing the loss of structural information. More examples of non-linear Bézier dissections are presented in Appendix B (please refer to the supplemental material).

5.2 Comparison Methodology of Protein Folding Patterns Using our Abstractions

As a *proof of concept*, we design a *simple-minded* strategy to compare and score the similarity between any two abstractions generated by our program.

A geometric profile is generated given any dissection of a protein structure. For every region in the dissection, each intermediate control point of the Bézier curve is projected onto the curve. This projection is the point on the curve that is closest to the control point. These projections are connected to the end points thus forming a series of line segments.

Any two skew lines have a mutually perpendicular vector. The angle required to rotate one line about this mutually perpendicular so that it eclipses the other line gives the orientation angle in the range [0, 360 degrees]. A table of orientation angles are recorded for each pair of line segments. In addition to the orientation angles, we record the distances between the mid-points of each pair of lines. The row-wise concatenation, of all possible orientation angles and the corresponding distances between lines from the table, results in a sequence of 2-tuples representing a simple geometric sequence profile of the protein. The basic idea here is that the sequences of geometric profiles of two proteins sharing similarity in their folding patterns would also be similar. This allows the use of the rich repertoire of sequence comparison methods to efficiently undertake large scale comparisons of protein folding patterns.

We align any two geometric profiles by implementing the standard affine-gap version of dynamic programming algorithm to compare pairs of sequences. An *ad hoc* scoring function is used to score matches in the alignment. Specifically the score of aligning any two angles w_i and w_j and the corresponding distances r_1 and r_2 between the midpoints of the interacting line segments is given by: $\text{score}(w_i, w_j) = (45^\circ - \Delta w) \exp(-\frac{\Delta r^2}{c^2})$, where $\Delta w = \min\{|w_i - w_j|, 360^\circ - |w_i - w_j|\}$, $\Delta r =$ difference of the two distances, and c is a constant which is set to 20 Å. With this new scoring function, we recompute the alignments.

The resultant alignment score derived by aligning any two geometric profiles is normalized as:

$$\hat{S}(A, B) = \frac{S(A, B)}{0.5 * (S(A, A) + S(B, B))}, \tag{3}$$

where $S(A, B)$ is the optimal score; $S(A, A)$ and $S(B, B)$ are the respective self-alignment scores. In summary, in order to compare any two Bézier segmentations, we generate their geometric profiles and do a sequence alignment of them.

5.3 Using the Bézier Segmentation Profiles for Fast Database Search

We show the applicability of Bézier segmentations by using them to achieve an efficient and accurate database search and retrieval. To do so we first use Bézier segmentations as a screening filter to identify a small candidate set of proteins from the database that are most likely to be structurally similar to the query. We then run sophisticated (and time consuming) structure alignment programs to accurately match the query with each of the candidate structures in this small set. Clearly, such an approach can save very significant amounts of computation time *without* losing accuracy, *if* the filtering method is effective. The following experiments assess the effectiveness of a filtering method based on Bézier segmentations.

We consider the entire ASTRAL-SCOP 40 (version 1.75) [19] database containing 11,146 structures. The database has 1,188 distinct folds and seven distinct classes. We note that no two domains in this data set share more than 40 percent amino acid sequence identity. Our experiment proceeded as follows. We first selected five structures *(queries)* by identifying the most common five folds types in the database (i.e., those present in more structures) and then randomly selecting a structure for each fold type. The queries used in our experiment are listed in Table 1. Once the five queries were

TABLE 1
Description of the Five Queries Selected

Fold type	Fold description	Query	Query domain
a.4	DNA/RNA binding-3-helical bundle	d2hosa_	a.4.1.1
b.1	Immunoglobulin-like β-sandwich	d1l6za1	b.1.1.1
c.1	TIM β/α-barrel	d1w0ma_	c.1.1.1
c.2	NAD(P)-binding Rossman-fold	d1gu7a2	c.2.1.1
d.58	Ferredoxin-like	d2fdna_	d.58.1.1

selected, the geometric profile of each query was aligned with the geometric profile of each of the 11,146 structures in the database, and the resulting alignment scores were used to rank the structures. Those structures at the top of the ranked list are expected to be the most similar to the query structure. We then filtered the top k percent (cutoff point) structures for varying values of $k(0, 5, 10, \ldots, 100)$ and considered them to have a fold identical to that of the query. Finally, we computed the *false positive rate (FPR)* and *true positive rate (TPR)* at each of these cutoff points and plotted them to generate a receiver operating characteristic (ROC) curve with which we establish the accuracy of the filtering method. The closer the curve is to the diagonal, the less accurate is the test. The area under the ROC curve (AUC) is, therefore, a direct measure of this accuracy.

TPR and FPR can be computed by evaluating the ranked list on the basis of fold similarity or class similarity. We calculate the metrics for both the cases and present the results in Table 2. The ROC curves for the query domains are shown in Fig. 7.

We observe that the accuracy of the test is greater when the discrimination between structures is at the fold level. The fold level discriminative ability provided by the Bézier segmentation profiles suggest that this method can be used to identify similar folds and hence, can be used in the initial screening of a database of structures. Once the candidate structures are filtered out, these can be scrutinized more thoroughly to find the structure with identical folding pattern as the query.

5.4 Comparison of Our Protein Alignment Method with Others

A quantitative assessment of the Bézier abstractions generated by our approach is demonstrated by undertaking a large scale comparison between protein structures abstracted using our approach. These structures are chosen such that they vary across an entire spectrum of

TABLE 2
Area Under the Curve Values when the Evaluation Is Done at the *Fold* and *Class* Levels

Query	AUC (fold level)	AUC (class level)
d2hosa_	0.87	0.74
d1l6za1	0.83	0.67
d1w0ma_	0.92	0.80
d1gu7a2	0.81	0.78
d2fdna_	0.82	0.58

structural relationships. Specifically, we randomly choose a data set containing 500 domains from the ASTRAL SCOP 40 (version 1.75) [19] database. We call these selected domains, *pivots*. For each pivot domain we randomly choose five distinct domains (differing in length by no more than 50 residues with respect to the pivot), such that the first belongs to the same SCOP family as the pivot, the second belongs to the same SCOP superfamily (but not family), the third belongs to the same SCOP fold (but not family or superfamily), the fourth belongs the same SCOP class (but not any better) and the fifth belongs to an entirely different SCOP class (the *decoy*).

Geometric profiles, from our non-linear Bézier curve abstraction, are generated for each of the $2{,}500\,(= 5 \times 500)$ SCOP domains in this data set. The geometric profiles of the 500 pivot domains are aligned separately with each of its five associated domains. A Box-Whisker plot is constructed to show the variance in the normalized alignment scores at each level in the SCOP hierarchy.

To serve as a gold standard, we align the same data set using the commonly used protein alignment methods (DALI [20], Matt [21], TM-align [22]). The value of the scoring function for each method is used to generate a Box-Whisker plot showing the variability under this scoring function. The plots are shown in Fig. 8.

A good discriminative scoring function should result in a box-whisker plot where the median values of alignment scores decreases monotonically as the structures being compared diverge in their evolutionary (structural) distance along the SCOP hierarchy: family, superfamily, fold, class, and decoy. Examining these plots, all the alignment methods show this behaviour. At the levels of SCOP family, superfamily and fold, the discrimination achieved using DALI and our approach remains comparable. However, as

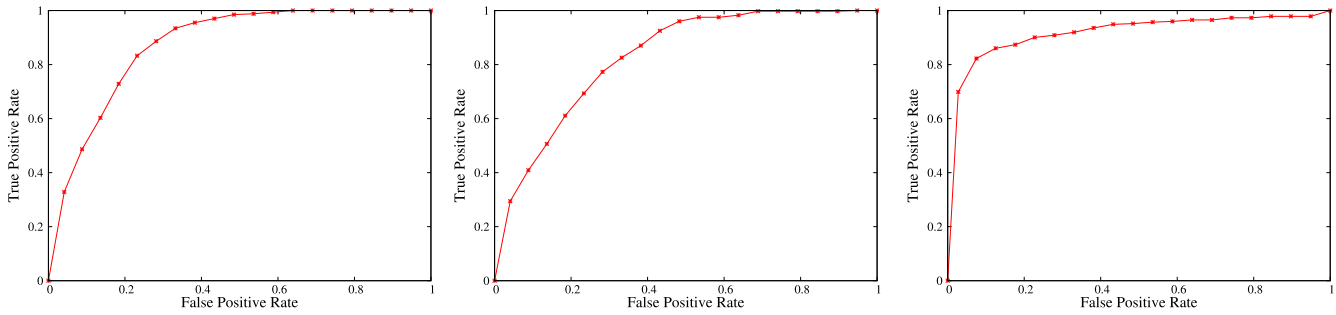

Fig. 7. ROC curves for the fold level evaluation. The corresponding AUC values are tabulated in Table 2. The red dots in each ROC curve denotes the (TPR,FPR) value at a cutoff points ($k = 0, 5, 10, \ldots, 100$).

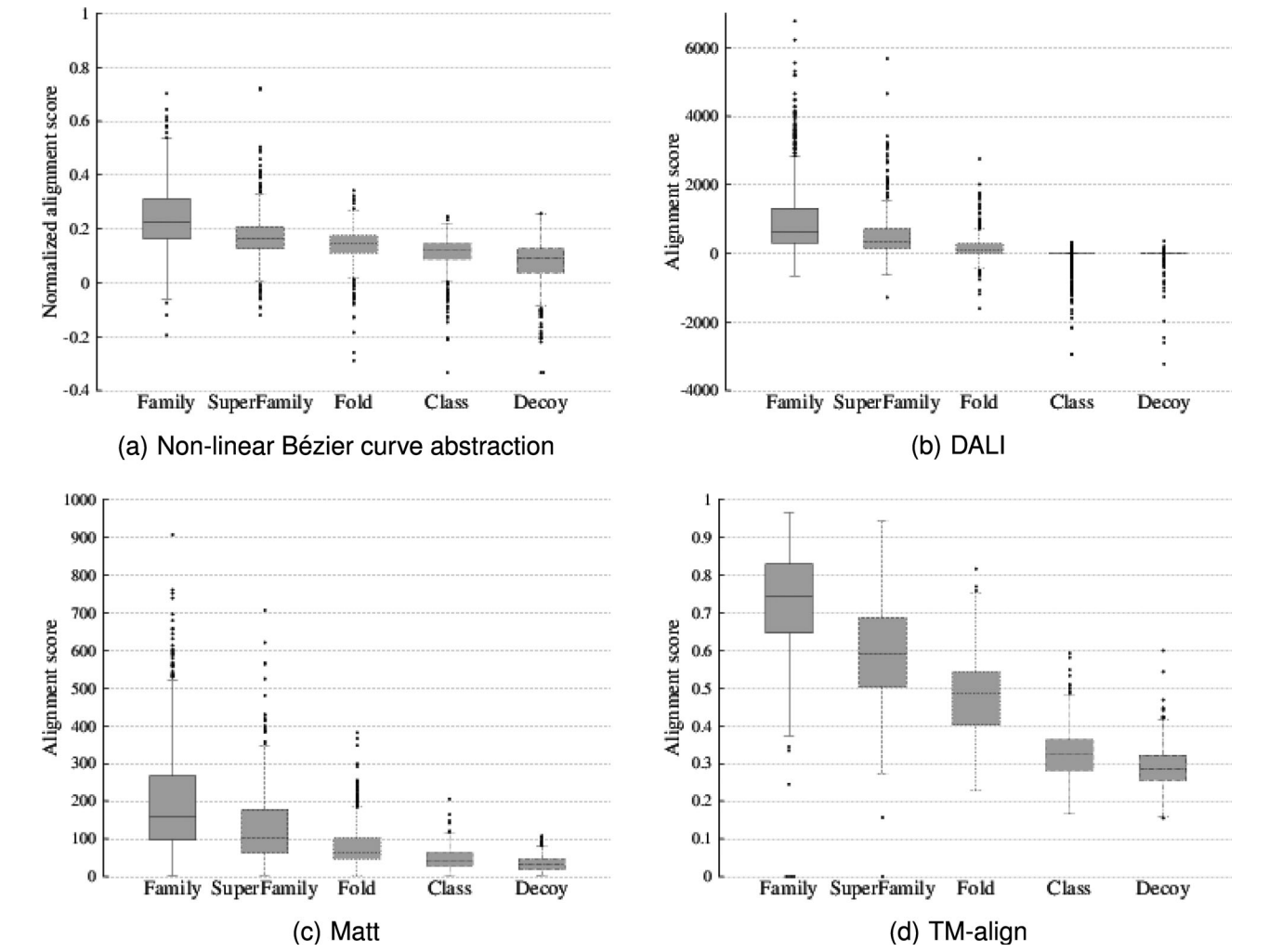

Fig. 8. Box-whisker plots of comparisons on a structural set derived from SCOP using the geometric profiles of our non-linear abstraction and other popular alignment methods *viz.* DALI, Matt, TM-align. (Note, the scale on the Y-axis differ between plots. A comparison of the widths of the boxes across two boxplots has no meaning. Each boxplot is used to assess the discriminative power across the SCOP hierarchy using an alignment method.)

expected of residue-residue comparisons, DALI is significantly better at the levels of Class and Decoy. This accuracy of DALI comes at a heavy computational cost. DALI takes about $\sim$12 hours to compute all the 2,500 alignments in the data set we considered. In contrast, it only takes us $\sim$15 minutes to compare the geometric profiles. This excludes the *one-time preprocessing* cost of computing the geometric profiles of the source structures in the collection.

We compared the standard scores of the normal distribution (or z-scores) for the DALI alignments with the z-scores for the raw alignment scores produced by our method. To compute the z-scores corresponding to our alignment scores, we considered the pairwise alignment scores at the class level to be the population values. The mean and standard deviation of these values are then calculated. These population parameters are then used in the computation of the z-score corresponding to a given alignment score. We then computed the correlation coefficient of the z-scores at different levels of the SCOP hierarchy. The results are tabulated in Table 3.

From the correlation coefficients, we observe that there is a significant linear dependence between the alignment z-scores produced by the two methods. This confirms the effectiveness of the alignments generated by our method at the family, superfamily, and fold level. However, at the class and decoy level, the degree of correlation is not substantial.

We also compared our alignments against the ones generated using TM-align and Matt. TM-align also provides a better discrimination at the Class and Decoy level. However, the variance of the alignment scores at each level of the SCOP hierarchy is greater compared to all other alignment methods. The alignment results of Matt and ours are also comparable as can be seen from the boxplots.

These results suggest that our abstraction allows for accurate fold-level discrimination using the simplest of the search strategies. The method of comparison is however used here only as a proof of concept. Our non-linear abstraction can be used as the basis of more rigorous methods for efficient large scale structure comparison. Towards this goal, we begin by exploring the use of Knot invariants from algebraic topology.

TABLE 3
Correlation of z-Scores of Alignments Obtained Using Our Method and DALI

Family	Superfamily	Fold	Class	Decoy
0.82	0.75	0.79	0.18	0.05

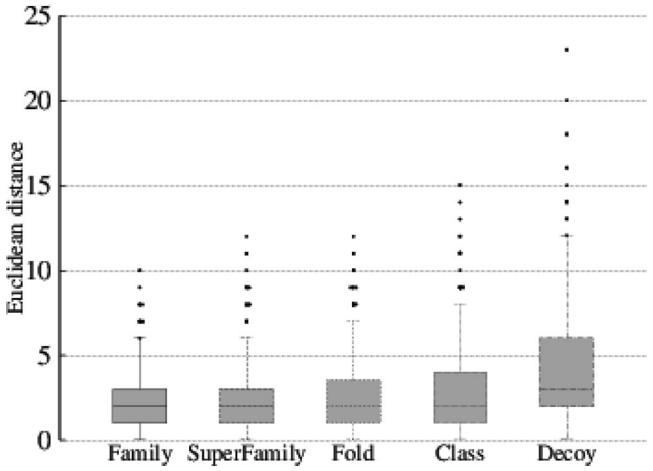

Fig. 9. Box-whisker plots for the non-linear Bézier curve abstractions compared using knot invariants (smaller value on the ordinate axis implies better structural relationships).

5.4.1 Knot Invariants Based Comparison

Røgen and Fain [23] introduced an approach to compare protein structures by representing each structure as a 30-dimensional vector of topological *knot invariants*. The similarity between any two structures is then given by the Euclidean distance between the two vectors. This forms an alignment-free methodology for structural comparison. The components of the vector correspond to the knot invariants computed by approximating the protein backbone as a polygon. Knot invariants are generic and provide a framework to compare non-linear curves. But because of the lack of closed form solutions in computing the individual invariants, the non-linear curves are approximated as polygons. There exists an analytical way to compute the invariants when the curve is comprised of line segments. The details are outlined in [23].

We adapt the idea to our approach. Each Bézier abstraction is approximated as a polygon by the method described in Section 5.2 (that constructs geometric profiles for comparison). Knot invariants are then computed using this polygon. We assess the discriminative ability on our data set selected from SCOP. These results are shown in Fig. 9. We notice that the discriminative ability using the knot invariant alignment-free approach is significantly poorer than the one achieved using simple geometric profiles (see Fig. 8a). We conjecture that to achieve a better discrimination using knot invariants, the Bézier curves should be considered raw (without abstracting them as polygons). This involves solving the Gaussian integrals which do not have closed form solutions (unlike the polygonal case considered in this exercise). This and other avenues of rigorous comparison will be pursued as a future direction of research.

6 Conclusion

The main goal of this paper was to develop a rigorous method to abstract protein folding patterns. We achieved this goal using the statistical and information theoretic framework of minimum message length inference. Our approach offers a robust mechanism to abstract protein structures using non-linear Bézier curves. This representation successfully captures the conformational plasticity commonly observed in protein structures thereby minimizing the loss of structural information, overcoming a major limitation with existing representations. Further, our representations can be used in various analyses involving protein structures. We demonstrated as a proof of concept the effectiveness of such abstractions for rapid structure comparison. A future direction of research will be to design an equally rigorous and efficient methodology for querying structural databases using our non-linear abstraction of protein folding patterns. Our program to dissect any given protein structure using non-linear Bézier curves is available from http://lcb.infotech.monash.edu.au/piecewise-nonlinear-fit/.

Acknowledgments

Parthan Kasarapu thanks NICTA for providing his PhD scholarship. NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

References

[1] A. M. Lesk, *Introduction to Protein Science: Architecture, Function, and Genomics*. Oxford, UK: Oxford Univ. Press, 2004.
[2] A. M. Lesk, "Systematic representation of protein folding patterns," *J. Molecular Graph.*, vol. 13, no. 3, pp. 159–164, 1995.
[3] C. Chothia and A. V. Finkelstein, "The classification and origins of protein folding patterns," *Annu. Rev. Biochemistry*, vol. 59, no. 1, pp. 1007–1035, 1990.
[4] C. Chothia, M. Levitt, and D. Richardson, "Helix to helix packing in proteins," *J. Molecular Biol.*, vol. 145, no. 1, pp. 215–250, Jan. 1981.
[5] F. E. Cohen, M. J. Sternberg, and W. R. Taylor, "Analysis of the tertiary structure of protein β-sheet sandwiches," *J. Molecular Biol.*, vol. 148, no. 3, pp. 253–272, 1981.
[6] N. Colloc'h, C. Etchebest, E. Thoreau, B. Henrissat, and J. Mornon, "Comparison of three algorithms for the assignment of secondary structure in proteins: The advantages of a consensus assignment," *Protein Eng.*, vol. 6, no. 4, pp. 377–382, 1993.
[7] W. R. Taylor, "Defining linear segments in protein structure," *J. Molecular Biol.*, vol. 310, pp. 1135–1150, 2001.
[8] A. S. Konagurthu, L. Allison, P. J. Stuckey, and A. M. Lesk, "Piecewise linear approximation of protein structures using the principle of minimum message length," *Bioinformatics*, vol. 27, no. 13, pp. i43–i51, 2011.
[9] C. S. Wallace and D. M. Boulton, "An information measure for classification," *Comput. J.*, vol. 11, no. 2, pp. 185–194, 1968.
[10] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, Jul. 1948.
[11] P. Elias, "Universal codeword sets and representations of the integers," *IEEE Trans. Inf. Theory*, vol. IT-21, no. 2, pp. 194–203, Mar. 1975.
[12] C. S. Wallace and P. R. Freeman, " Estimation and inference by compact coding," *J. Royal Stat. Soc. Series B (Methodological)*, vol. 49, no. 3, pp. 240–265, 1987.
[13] C. S. Wallace, *Statistical and Inductive Inference Using Minimum Message Length*, series Information Science and Statistics. New York, NY, USA: Springer Verlag, 2005.
[14] Y. W. Chen, T. Tajima, and S. Agrawal, "The crystal structure of the ubiquitin-like (UbL) domain of human homologue A of Rad23 (hHR23A) protein," *Protein Eng. Des. Sel.*, vol. 24, no. 1–2, pp. 131–138, 2011.
[15] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, vol. 22, pp. 2577–637, Dec. 1983.
[16] S. Shi, B. Chitturi, and N. V. Grishin, "ProSMoS server: A pattern-based search using interaction matrix representation of protein structures," *Nucleic Acids Res.*, vol. 37, no. suppl 2, pp. W526–W531, 2009.

[17] A. S. Konagurthu, P. J. Stuckey, and A. M. Lesk, "Structural search and retrieval using a tableau representation of protein folding patterns," *Bioinformatics*, vol. 24, no. 5, pp. 645–651, 2008.
[18] A. S. Konagurthu, A. M. Lesk, and L. Allison, "Minimum message length inference of secondary structure from protein coordinate data," *Bioinformatics*, vol. 28, no. 12, pp. i97–i105, 2012.
[19] A. Murzin, S. Brenner, T. Hubbard, and C. Chothia, "SCOP: A structural classification of proteins database for the investigation of sequences and structures," *J. Molecular Biol.*, vol. 247, no. 4, pp. 536–540, 1995.
[20] L. Holm and C. Sander, "Protein structure comparison by alignment of distance matrices," vol. 233, no. 1, pp. 123–138, 1993.
[21] M. Menke, B. Berger, and L. Cowen, "Matt: Local flexibility aids protein multiple structure alignment," *PLoS Computat. Biol.*, vol. 4, no. 1, p. e10, 2008.
[22] Y. Zhang and J. Skolnick, "TM-align: A protein structure alignment algorithm based on the TM-score," *Nucleic Acids Res.*, vol. 33, no. 7, pp. 2302–2309, 2005.
[23] P. Røgen and B. Fain, "Automatic classification of protein structure by using Gauss integrals," *Proc. Nat. Acad. Sci.*, vol. 100, no. 1, pp. 119–124, 2003.

Parthan Kasarapu received the BTech and MTech (2011) degrees in computer science & engineering from the Indian Institute of Technology, Madras, India. He is currently working toward the PhD degree in computer science at the faculty of information technology at Monash University, Australia. His present research focus is on computational modeling of protein three-dimensional structure and architecture using the principle of minimum message length, a method of Bayesian model inference.

Maria Garcia de la Banda is a professor and the deputy dean of Monash University's Faculty of Information Technology. Her current research interests are in modeling, analysis, and transformation of combinatorial and satisfaction optimization problems (common in the transport, logistics and health areas) as well as in bioinformatics. She is an area editor of the *Journal of Theory and Practice of Logic Programming*. She has been an elected member of the Executive Committee of the Association of Logic Programming, and was the program co-chair of the 2008 International Conference in Logic Programming.

Arun S. Konagurthu is a senior lecturer at Monash University's Faculty of Information Technology. He leads the faculty's research flagship on computational biology. His research interests include bioinformatics, discrete data structures and algorithms, combinatorial optimization, information theory, and statistical inference.