Using 3D Hidden Markov Models that explicitly represent spatial coordinates to model and compare protein structures

Vadim Alexandrov[1*] & Mark Gerstein[1,2]

[1]*Department of Molecular Biophysics and Biochemistry,*
[2]*Department of Computer Science,*
Yale University,
266 Whitney Ave., New Haven, CT 06511, USA

Phone: (203) 432-6105
Fax: (203) 432 5175

E-mail:
Vadim Alexandrov: vadim.alexandrov@yale.edu
Mark Gerstein: mark.gerstein@yale.edu

[*] To whom correspondence should be addressed: *Vadim.Alexandrov@Yale.Edu*

Abstract

Background

Hidden Markov Models (HMMs) have proven very useful in computational biology for such applications as sequence pattern matching, gene-finding, and structure prediction. Thus far, however, they have been confined to representing 1D sequence (or the aspects of structure that could be represented by character strings).

Results

We develop an HMM formalism that explicitly uses 3D coordinates in its match states. The match states are modeled by 3D Gaussian distributions centered on the mean coordinate position of each alpha carbon in a large structural alignment. The transition probabilities depend on the spread of the neighboring match states and on the number of gaps found in the structural alignment. We also develop methods for aligning query structures against 3D HMMs and scoring the result probabilistically. For 1D HMMs these tasks are accomplished by the Viterbi and forward algorithms. However, these will not work in unmodified form for the 3D problem, due to non-local quality of structural alignment, so we develop extensions of these algorithms for the 3D case. Several applications of 3D HMMs for protein structure classification are reported. A good separation of scores for different fold families suggests that the described construct is quite useful for protein structure analysis.

Conclusion

We have created a rigorous 3D HMM representation for protein structures and implemented a complete set of routines for building 3D HMMs in *C* and *Perl*. The code is freely available from http://www.molmovdb.org/geometry/3dHMM , and at this site we also have a simple prototype server to demonstrate the features of the described approach.

Introduction

HMMs have been enormously useful in computational biology. However, they have only been used to represent sequence data up to now. The goal of the present work is to make HMMs operate fundamentally with *3D-structural* rather than *1D-sequence* data. Since HMMs have proven worthwhile in determining a characteristic profile for an ensemble of related sequences, we expect them to be useful in building a rigorous mathematical description of protein fold family. Our work rests on three elements of background theory: 1D HMMs, 3D structural alignment and 3D core structures.

One-dimensional HMMs

Profile hidden Markov models (profile HMMs) are statistical models of the primary structure consensus of a sequence family. Krogh et al [1] introduced profile HMMs to computational biology to analyze amino acid sequence similarities, adopting HMM techniques that had been used for years in speech recognition [2]. This paper had a propelling impact, because HMM principles appeared to be well suited to elaborating upon the already popular "profile" methods for searching databases using multiple alignments instead of single query sequences [3]. In this context an important property of HMMs is their ability to capture information about the degree of conservation at various positions in an alignment and the varying degree to which indels are permitted. This explains why HMMs can detect considerably more homologues compared to simple pairwise comparison [4, 5]. Since their initial use in modeling sequence consensus, HMMs have been adopted as the underlying formalism in a variety of analyses. In particular, they have been used for building the Pfam database of protein familes [6-8], for gene finding [5], for predicting secondary structure [9] and transmembrane helices [10]. Efforts to use sequence-based HMMs for protein structure prediction [11], fold/topology recognition [12-14] and building structural signatures of structural folds [15] were also reported recently. However, no one yet has built an HMM that explicitly represents a protein in terms of 3D coordinates. A further key advantage of using HMMs is that they have a formal probabilistic basis. Bayesian theory unambiguously determines how all the probability (scoring)

parameters are set, and as a consequence, HMMs have a consistent theory behind gap penalties, unlike profiles.

A typical HMM (see Figure 1) consists of a series of states for modeling an alignment: *match* states M_k for consensus positions; and *insert* I_k and *delete* states D_k for modeling insertions/deletions relative to the consensus. Arrows indicate state-to-state transitions, which may occur according to the corresponding transition probabilities. Sequences of states are generated by the HMM by following a path through the model according to the following rules:

- The path is initiated at a begin state M_0; subsequent states are visited linearly from left to right. When a state is visited, a symbol is output according to the emission probability of that state. The next state is visited according to current state's transition probabilities.
- The probability of the path is the product of probabilities of the edges traversed. Since the resulting sequence of states is observed and underlying path is not, the part of the HMM considered "hidden" is the path taken through the model.

Structural alignment

Structural alignment involves finding equivalences between sequential positions in two proteins (Figure 2). As such, it is similar to sequence alignment. However, equivalence is determined on the basis of a residue's 3D coordinates, rather than its amino acid "type." A number of procedures for automatic structural alignment have been developed [16-24]. Some of these are based on iterative applications of dynamic programming, where each iteration minimizes the *RMS* distances between the newly aligned atoms. Others maximize the overlap of distance matrices, and yet others are based on heuristics such as hashes. Structural alignment has been used to find non-obvious similarities in protein structure -- e.g. the globin-colicin similarity [25] -- to cluster the whole structure databank [26-28], to refine measures of structural annotation transfer [29], and to assist homology modeling [30].

The next step after pairwise structural alignment is multiple structural alignment, simultaneously aligning three or more structures together. There are currently a number of approaches for this [18, 23, 31, 32]. Most of these proceed by analogy to multiple sequence alignment [33-35], building up an alignment by adding one structure at a time to the growing consensus. Multiple structural alignment is an essential first step in the construction of *consensus* structural templates, which aim to encapsulate the information in a family of structures. It can also form the nucleus for a large multiple sequence alignment -- i.e., highly homologous sequences can be aligned to each structure in the multiple alignment.

Core structures

Given a structural alignment of a family of proteins, a set of atoms with essentially fixed relative positions in all member of the family can be determined [36, 37]. This set is called the *invariant core* for the given family. Cores are meant to characterize protein families statistically, based on positional variation in observed main chain atoms among members of structural family.

Computing a core structure involves the following steps: Superimposition (*structural alignment*) of an ensemble of protein structures from the same family, calculation of structural deviation between coordinates from ensemble structures, iterative removal of non-core atoms based on high positional variation, calculation of ellipsoid volumes representing positional variance of alpha carbon positions at every core center.

Difficulty in adapting 1D HMM Formalism to 3D

There have been a number of recent attempts to make HMMs be useful for structural studies [9, 10]. However, none of the suggested schemes are fundamentally three-dimensional (coordinate dependent), since all of them are based on building a 1D HMM profile representing a sequence alignment and structural information only enters in the form of encoded symbols (i.e. *H* for helix and *E* for sheet). Adding in real 3D structure is non-trivial. This reflects the fact that the structure is fundamentally different from the sequence not only in increased dimensionality, but also due to the transition from discrete to continuous representation. For matching a query structure of the model, the conventional

dynamic programming that underlies normal HMMs will not work since structural alignment is "non-local". Normal dynamic programming assumes that determining the best match between query and model in a given "local" region of the alignment potentially will not affect the optimum match sequentially *before* this region. Thus, one can break up the whole sequence comparison problem into sub-problems that can be readily solved. However, this does not apply in structure comparison. The optimum match in one region of the alignment R potentially can affect the optimum superposition between two structures and this in turn can affect the optimum alignment globally, in regions sequentially distant from R.

Combining 1D HMMs with 3D Cores to construct 3D HMMs

To get around this difficulty, we develop a way of doing an alignment-free superposition initially and then adding conventional HMM scoring. In our approach, the core structures are made from "ellipsoidal" Gaussian distributions centered on aligned Cα positions. If each Gaussian distribution is normalized to 1, we have a probability distribution based on coordinates. Hence, if we want to use HMMs on protein structures instead of sequences, core models provide a nice representation for the match states. In fact, we can think of the cores as "structural profiles", with each core ellipsoid representing a statistical distribution of potential coordinates, just as in sequence profiles, each match state represents a probability distribution of each of the 20 amino acids occurring in that position. In practice for a sequence profile, state emission distributions correspond to tables of probabilities of each amino acid appearing. Likewise, in 3D HMMs match states correspond to the probability of a given Cα position falling within a prescribed volume. This can be readily calculated from coordinate differences: if an aligned Cα from the query appears close to the ellipsoid centroid, it scores well; if it is farther away, it scores poorly.

A path through a HMM is a sequence of states such that there exists an edge from each state in the path to the next state in the path. A path through the 3D HMM gives a probability distribution of each structural position, based on the probabilities of the atom positions in the corresponding states. The probability of the structure given a core target is the product of the probabilities of the Cα positions in each state. The

probability of an HMM generating a structure is the sum, over all the paths in the HMM, of the probability of the path times the probability of the structure given the path.

Transition probabilities of the new model are based on the probabilities of the coordinates of a given atom within a coordinate distribution (ellipsoid), where the probabilities are calculated based on the relative distance of the query atom from each ellipsoid centroid. For regions not associated with the core transition probabilities are derived from a multiple alignment in a similar fashion to those in 1D sequence HMMs, and we employ a Bayesian approach to merge them together.

We, therefore, suggest a combined approach. A separate HMM is computed for each core structure we want to use. Every core ellipsoid corresponds to the match state in the HMM. The sizes of the ellipsoids are related to the HMM transition probabilities: the larger size corresponds to a greater indel probability, whereas the smaller one indicates a more conserved core position and thus increased probability of transition to the corresponding match state (and a lower probability of falling into an indel state). Next, a query protein structure is aligned with a core and relative interstructural distances are computed. These deviations between the query and the core model can be converted to the emission probabilities in the match states of the HMM, and an overall score can finally be computed. This score represents the log likelihood that the protein structure was generated by the HMM, and not some other disordered model, called the null model. In practical terms, this score can be used to identify the new protein as a potential member of a protein family, and thus classify it.

Results and Discussion

To demonstrate the power of our three-dimensional probabilistic constructs for protein structure analysis, we performed the following calculations. First, we built 3D HMMs for globin and IgV fold families from the sets of aligned representative structures. Eighteen structures were chosen to create a 3D HMM for the globin family and four structures to represent the IgV domains. We selected a small number of representatives for the IgV HMM in an attempt to establish a statistically meaningful threshold for the

minimal size of a typical training set. We scored all available globin and IgV SCOP domains against our constructed models. The separation of scores appeared to be quite good (Figure 4). Globin 3D HMM and IgV 3D HMM turned out to be capable of distinguishing the queries from its own class with 98.2% and 96.6% accuracy, respectively. We explain minor overlaps of the scores by incompleteness of the chosen training sets as well as by our inability to achieve optimal superposition at the structural alignment stage for certain cases. To the contrary, the corresponding RMSD values calculated for the alignment of all IgV's and globin domains against the IgV core (match states from the IgV 3D HMM) resulted in quite significant overlap and appeared to be much less informative for structural classification (Figure 5). We use RMSD values as a benchmark because the procedure involving structural alignment of the query against the core remains the same in both RMSD and 3D HMM calculations, whereas the improvement of the 3D HMM scores separation comes exclusively from the HMM scoring scheme.

To further test performance of our method we performed more challenging, (α/β) vs (α/β) and (α+β) vs (α+β) classification-type calculations: FAD/NAD(P)-binding domains (SCOP fold 51904) vs NAD(P)-binding domains (SCOP fold 51734), Flavodoxin fold (SCOP fold # 52171) vs Thioredoxin fold (SCOP fold # 52832) and Ferrodoxin fold (SCOP fold # 54861) vs Lysozyme fold (SCOP fold # 53954). Each model was built by choosing from five to ten non-identical (less than 95% sequence identical) representatives, which had RMSD no greater than 3 Angstroms (within the set) and the minimal available length of sequence consensus. Insert-insert transition probabilities $p_{I_0 \to I_0}$ and $p_{I_N \to I_N}$ were manually set equal to 0.95 in order to minimize the insertion penalties at the terminal ends of the model (to model the flanking ends of the longer structures which didn't enter the training set). Thus, our "minimal core" models effectively scored the best superimposed part of the query with respect to the core centroids. Note, that in the corresponding RMSD calculation, the flunking ends do not affect the RMSD value, because only the distances for the determined query $\leftrightarrow$ model equivalence pairs enter the score. Overlap of the 3D HMM log scores for these calculations never exceeded 15% in the calculations (Figures 6a-6c), whereas the RMSD scores overlapped more than 65% (not shown).

We also scored all 95% or less sequence-identical domains against the 3D HMM for Thioredoxin fold (Figure 7). One can readily see that the scores for all non-Thioredoxin folds are located mostly to the right of the scores typical for the Thioredoxin fold.

Finally, we performed a different classification-type calculation: given a 3D HMM built for the ferrodoxin family from four representative structures, we scored the same globin and IgV domains against it. Even though *none* of the queries belonged to the ferrodoxin 3D HMM, the separation of HMM scores for the two folds was still as high as in the previous calculations, namely 97.5%. This suggests that a given model may be useful not only for binary classification but also for dividing up fold space in more distance oriented terms. On the other hand, the corresponding RMSD values ranged from 4 to 11 Angstroms with no detectable separation at all. This can be readily understood: a 3D HMM is designed in such a way that different spatial regions are assigned different *spatial penalties* in the form of the coordinate-dependent emission, insertion and deletion probabilities. Therefore, when aligned with the core of the ferrodoxin HMM, queries from different families would encounter different and very specific for each family score penalties for appearing in the "wrong", family-specific regions of space. RMSD values for such alignments can be almost identical for the two families; however, the corresponding HMM scores will, most probably, appear confined within relatively small and, in general, separable intervals. In contrast, two queries, one with seemingly large RMSD score and another one with a much smaller value, may still rightfully belong to the same 3D HMM class, provided that the first query has its high RMSD structural parts appearing in the "allowed" (low-penalty) spatial regions. With help of this highly useful 3D HMM feature, one should be able to develop a scale of scores for various structural groups of interest, which can be further used for classification of the unknown structures (or parts of structures) in the same manner as chemical shifts for different functional groups are used in NMR structure determination experiments.

Conclusion and future directions

We have created a rigorous three-dimensional HMM representation for protein structures. This incorporates the major elements of HMM theory: a set of directionally connected match, insert, and delete states, transition and emission probabilities, the optimal Viterbi path, and forward algorithm. We have not yet considered such aspects as using Expectation maximization and the Baum-Welch procedure for optimization of model parameters. However, currently our procedure is self-contained and ready for protein structure analysis. Preliminary results indicate that the new constructs can be quite useful for protein structure analysis and classification.

Methods

In outline our procedure consists of the following steps:

1. 3D representation for the match states and emission probabilities;

2. A procedure for superposing a query against the match states without consideration of transitions;

3. Calculation of transition probabilities from the core parameters and evaluation of corrections from observed gaps in a multiple alignment;

4. Modified forward and Viterbi algorithms for calculating the probability that the query was generated from the 3D HMM and calculating the best alignment of the query to the model.

A conventional 1D HMM would comprise steps 3 and 4. However, because we are dealing with 3D coordinates a single application of the dynamic programming in the Viterbi sense is not sufficient to match a query against the model. The reason is that changing locally the match between query and model affects the global overall superposition of the structure. Hence, we have to find a superposition first in heuristic fashion without consideration of the transitions before scoring with transitions. To achieve this

optimal superposition, we align the centers of the ellipsoids (match states in the model) with the Cα coordinates of the query.

Three-dimensional core representation for HMM match states

The parametric representation of the core structure is built by modeling a distribution of atomic positions as a 3D Gaussian. Having a set of structures, superimposed using an RMS criteria, we obtain an alignment, which pairs each atom k in one structure with an equivalent atom in the others. The mean position $\boldsymbol{r}_k \equiv (x_k, y_k, z_k)$ and 3D-covariance matrix for each such center k can be calculated:

$$\boldsymbol{\sigma}^{(k)} = \begin{pmatrix} \sigma^{(k)}_{x^2} & \sigma^{(k)}_{xy} & \sigma^{(k)}_{xz} \\ \sigma^{(k)}_{yx} & \sigma^{(k)}_{y^2} & \sigma^{(k)}_{yz} \\ \sigma^{(k)}_{zx} & \sigma^{(k)}_{zy} & \sigma^{(k)}_{z^2} \end{pmatrix}$$

We can now use this statistical representation $\left\{\boldsymbol{r}_k, \boldsymbol{\sigma}^{(k)}\right\}_{k=1}^{L_m}$ to position a query structure $\left\{\boldsymbol{q}_m \equiv (\mathrm{x}_m, \mathrm{y}_m, \mathrm{z}_m)\right\}_{m=1}^{L_q}$ with respect to the centers of the obtained core structure. The modified relative distances $\boldsymbol{r}'_{km} \equiv \boldsymbol{R}_k \cdot (\boldsymbol{r}_k - \boldsymbol{q}_m) = (x'_{km}, y'_{km}, z'_{km})$ of the *m*-th query atom with respect to the *k*-th core position are obtained by applying the rotation matrix $\boldsymbol{R}_k$, which diagonalizes the covariance matrix $\boldsymbol{\sigma}^{(k)}$,

$$\boldsymbol{D}^{(k)} = \boldsymbol{R}_k^T \boldsymbol{\sigma}^{(k)} \boldsymbol{R}_k,$$

where $\boldsymbol{D}^{(k)}$ is a diagonal matrix whose diagonal elements $d^{(k)}$ are the variances of a 3D distribution of the *k*-th ellipsoid that is oriented along the global coordinates. Clearly, core ellipsoids represent the emission probability distributions for the match states, i.e. the probability that a query atom is emitted by the match state located at the center of the corresponding ellipsoid.

Since the protein backbone is not continuous we have to discretize our emission probability distributions in order to obtain finite values for the observed (emitted), finite series of the query atoms. In the present work we choose variances $d^{(k)}_{x^2}, d^{(k)}_{y^2}$ and $d^{(k)}_{z^2}$ to make this discretization. Figure 3 illustrates discretization of a Gaussian distribution in one-dimensional case. Each distribution is partitioned amongst

intervals equal to the corresponding variance value, i. e. [0 - $d^{(k)}_{\alpha^2}$, $d^{(k)}_{\alpha^2}$ -2 $d^{(k)}_{\alpha^2}$,2 $d^{(k)}_{\alpha^2}$ -3 $d^{(k)}_{\alpha^2}$,…,$N d^{(k)}_{\alpha^2}$ - ∞], $\alpha = x, y, z$. The values of the distribution integrated over each interval give the desired values for the discretized emission probability distribution:

$$p_\alpha^{(i)} = \frac{1}{\sqrt{2\pi} d_{\alpha_k \alpha_k}} \int_{i \cdot d_{\alpha_k \alpha_k}}^{(i+1) \cdot d_{\alpha_k \alpha_k}} e^{-\frac{\alpha^2}{2 d^2_{\alpha_k \alpha_k}}} d\alpha, \quad i = 1..N-1$$

$$p_\alpha^{(N)} = \frac{1}{\sqrt{2\pi} d_{\alpha_k \alpha_k}} \int_{N \cdot d_{\alpha_k \alpha_k}}^{\infty} e^{-\frac{\alpha^2}{2 d^2_{\alpha_k \alpha_k}}} d\alpha$$

.

In practical computer implementation the integrals are replaced by numerical values of the gamma function $\Gamma(a)$ and incomplete gamma function $\gamma(a, \alpha) \equiv \frac{1}{\Gamma(a)} \int_\alpha^\infty e^{-t} t^{a-1} dt$ by using relationship

$$\int_0^\alpha e^{-t^2} dt = \frac{\sqrt{\pi}}{2} \gamma(\frac{1}{2}, \alpha^2) .$$

Note that our discretization procedure effectively partitions the volume of the *k*-th ellipsoid $V_k = \frac{4\pi}{3} d^{(k)}_{x^2} d^{(k)}_{y^2} d^{(k)}_{z^2}$, the measure of the penalty for the deviation from the ellipsoid center, creating a discrete representation of the emission probability distribution for each match state *k*. The *observed* emission probabilities (scores) e_{kn} -- i.e. those resulting from the alignment of the query with the core -- will be given as a product of the corresponding discretized emission probabilities for each coordinate:

$$e_{kn} = \prod_\alpha \varepsilon^{(\alpha)}_{kn} \ , \ \alpha = x, y, z .$$

The component emission probabilities $\varepsilon^{(\alpha)}_{kn}$ are determined from following relation:

$$\varepsilon^{(\alpha)}_{kn} \equiv p^{(i)}_\alpha (d_{\alpha_k \alpha_k}) \left\{ i : i \cdot d_{\alpha_k \alpha_k} < \hat{\boldsymbol{I}}^{(\alpha)} \boldsymbol{r}'_{kn} < (i+1) \cdot d_{\alpha_k \alpha_k} \right\},$$

where operator $\hat{\boldsymbol{I}}^{(\alpha)}$ projects out component α from vector $\boldsymbol{r}'_{kn}$.

Calculation of transition probabilities from core parameters and evaluation of corrections from alignment gaps

In addition to setting the values of the match emission values, volumes of the core ellipsoids can help parameterize the values for the transitions. A larger ellipsoid volume is associated with a greater chance of having an indel nearby, thus decreasing the probability of going into the corresponding match state. Therefore, it is logical to assume that the probability of transition to the match state should be inversely proportional to the volume of the corresponding core ellipsoid. In the present work we suggest the following values for priors on the transition probabilities:

$$p_{\forall \to M_k} = \frac{1}{1+V_k},$$
$$p_{\forall \to I_k} = p_{\forall \to D_k} = \frac{1}{2}\left(1 - \frac{1}{1+V_k}\right),$$

where $\forall$ denotes any state in the model from where transition is allowed by the inherent HMM topology. One can easily see that the above formulas produce higher penalties for transitions deviating from highly conserved core positions (match states).

We use the observed pattern of gaps in a known multiple sequence alignment as corrections to the above formulas for the transition probabilities. In particular, gaps in the sequence alignment corresponding to the non-core regions designate spaces with the higher insertion probabilities, whereas missing atoms in each core position is a sign of increasing deletion probability. The sequence alignment corrections $p'_{\forall \to \forall'}$ ($\forall$ representing any of the I_k, D_k or M_k states) can be introduced by direct application of pseudocount technique, which is easily formalized in Bayesian framework [7]:

$$p'_{\forall_{k-1} \to I_k} = \frac{p_{\forall_{k-1} \to I_k} \cdot n_0 + (M \cdot L_{\Delta_k} - N_{\Delta_k})}{n_0 + M \cdot L_{\Delta_k}},$$
$$p'_{\forall_{k-1} \to M_k} = 1 - p'_{\forall_{k-1} \to I_k}$$

where M is the number of structures in the (training) multiple alignment; N_{Δ_k} is the total number of amino acid symbol absences within a gap immediately preceding a particular HMM match state *k*, L_{Δ_k} is the length of the gap, and n_0 is a so-called pseudocount constant [7], which scales the correction. In the above formula, $M \cdot L_{\Delta_k}$ represents the total number of symbols within the gap (counting both amino acid symbols and amino acid absence symbols), and $M \cdot L_{\Delta_k} - N_{\Delta_k}$ gives the number of amino acid symbols present in the gap. If the pseudocount constant n_0 is set equal to 1, the above formula transforms to a well-known Laplace rule [7].

The corresponding corrections resulting from the gaps observed at actual HMM match states are calculated in the same manner

$$p'_{\forall_{k-1}\to D_k} = \frac{n_0 \cdot p_{\forall_{k-1}\to D_k} + N_{\Delta_k}}{n_0 + M},$$
$$p'_{\forall_{k-1}\to I_k} = p'_{\forall_{k-1}\to M_k} = 1 - p'_{\forall_{k-1}\to D_k}$$

The final value for the transition probability is the renormalized sum of the priors and the corresponding corrections:

$$\tilde{p}_{\forall\to\not\forall} = \frac{p_{\forall\to\not\forall} + p'_{\forall\to\not\forall}}{\sum_{\forall} p_{\forall\to\not\forall} + \sum_{\forall} p'_{\forall\to\not\forall}},$$

where $\not\forall$ denotes any HMM state accessible from state $\forall$. The optimization of the obtained probabilities with respect to maximization of the scores for the training sets may be necessary for the final tune-up.

Viterbi and Forward algorithms for aligning and scoring the query

Once one has two structures (the query and the centers of the core ellipsoids) optimally aligned in a gap-independent fashion and has a set of transition probabilities calculated between the match states, one can use standard HMM methods to calculate the single best alignment between the query and the model (the

Viterbi algorithm) and the overall score for matching the query to the model, integrated over all possible paths through the model (the forward algorithm). Our implementation of the Viterbi algorithm is straightforward, following that in Durbin et al. [7]. However, we rederived the Forward Algorithm equations [2] for the case of fully interconnected HMM using matrix calculus notations. This made the computer implementation more straightforward and faster for structures. Our forward algorithm equations assume the following form

$$\boldsymbol{f}_1 = \boldsymbol{\pi} \circ \boldsymbol{e}_1 , \qquad \text{(initiation)}$$

$$\boldsymbol{f}_{l+1} = \boldsymbol{e}_l \circ \boldsymbol{A}^T \boldsymbol{f}_l , \quad l = 1..L_q - 1 , \qquad \text{(recursion)}$$

$$P = \boldsymbol{1} \cdot \boldsymbol{f}_{L_q} , \qquad \text{(termination),}$$

where $\boldsymbol{\pi}$ denotes the initial probability distribution among the possible begin states, $\boldsymbol{A}$ is the transition matrix of dimension L_m x L_m, $\boldsymbol{e}_l$ is the vector of emission probabilities for the state *l*, $\boldsymbol{f}_l$ is the vector of forward variables in the *l*-th step, $\boldsymbol{1}$ is the vector with components all equal to 1, L_q is the length of the observation sequence (query), and P is the total probability that the observation sequence was emitted by the given model. Note the distinction between the scalar product $\mathbf{x} \cdot \mathbf{y} \equiv \sum_i x_i y_i$ and vector multiply operation (Hadamard product) $\mathbf{x} \circ \mathbf{y} \equiv \{x_i y_i\}_{i=1}^{Dim\{\mathbf{x}\}}$ that are related as $\mathbf{x} \cdot \mathbf{y} = \mathbf{1} \cdot (\mathbf{x} \circ \mathbf{y})$.

For the bigger HMMs with number of states 100 and higher, direct application of the above formulas involves many multiplications of small numbers (transition and emission probabilities) and usually results in severe underflow problems in practical computer implementations. These can be avoided if one works with logarithms of the involved quantities, even when one needs to perform addition or subtraction. Furthermore, because of the directional nature of the HMM topology many transition probabilities (those between non-adjacent states) are identically zero. Therefore, instead of dealing with the full L_m x L_m transition matrix $\boldsymbol{A}$ we can work with three *3*x L_m submatrices a_M, a_D and a_I, which

represent transitions from the match, delete and insert states respectively. The formulas for the HMM scoring based on the forward algorithm then take the form (with notation identical to that used in [7]):

Initialization:

$$f_{M_0}^{(0)} = 1; f_{M_k}^{(0)} = 0; f_{I_k}^{(0)} = 0; k > 0,$$
$$f_{D_1}^{(0)} = a_{M_0 \to D_1}; f_{D_k}^{(0)} = a_{M_0 \to D_1} \cdot \prod_{j=1}^{k-1} a_{D_j \to D_{j+1}}, k > 1,$$
$$f_{I_0}^{(i)} = e_{I_0}^{(i)} \cdot (f_{M_0}^{(i-1)} \cdot a_{M_0 \to I_0} + f_{I_0}^{(i-1)} \cdot a_{I_0 \to I_0}), i = 1..M_Q$$

Recursion:

$$\log(f_{M_k}^{(i)}) = \log(e_{M_k}^{(i-1)}) + \log(f_{M_{k-1}}^{(i-1)} \cdot a_{M_{k-1} \to M_k} + f_{I_{k-1}}^{(i-1)} \cdot a_{I_{k-1} \to M_k} + f_{D_{k-1}}^{(i-1)} \cdot a_{D_{k-1} \to M_k})$$
$$\log(f_{I_k}^{(i)}) = \log(e_{I_k}^{(i)}) + \log(f_{M_k}^{(i-1)} \cdot a_{M_k \to I_k} + f_{I_k}^{(i-1)} \cdot a_{I_k \to I_k} + f_{D_k}^{(i-1)} \cdot a_{D_k \to I_k})$$
$$\log(f_{D_k}^{(i)}) = \log(f_{M_{k-1}}^{(i)} \cdot a_{M_{k-1} \to D_k} + f_{I_{k-1}}^{(i)} \cdot a_{I_{k-1} \to D_k} + f_{D_{k-1}}^{(i)} \cdot a_{D_{k-1} \to D_k})$$

Termination:

$$\log(P) = \log(f_{M_{L_m}}^{L_q} \cdot a_{M_{L_m} \to M_{L_m+1}} + f_{I_{L_m}}^{L_q} \cdot a_{I_{L_m} \to M_{L_m+1}} + f_{D_{L_m}}^{L_q} \cdot a_{D_{L_m} \to M_{L_m+1}})$$

Our initialization of $f_{D_k}^{(0)}$ is different from $f_{D_k}^{(0)} = 0$, which is given in [7] for a similar HMM profile. A detailed derivation of the correct initialization condition along with accompanying examples will be published elsewhere.

Figures

Figure 1: Typical 1D HMM topology (adapted from [7])

Figure 2: Structural alignment of two protein fragments (backbones). Aligned parts are shown in yellow.

Figure 3: Discretization of the coordinate probability distribution in one dimension.

Figure 4. Separation of scores for IgV (red) and globin (blue) domains scored against globin 3D HMM.

Figure 5. Histogram of RMSD values for globin vs IgV domains and their overlap.

RMSD values were calculated for the alignment of these domains against the globin core. RMSD histogram scores for globin domains are shown in yellow, for IgV domains in blue and their overlap in green.

Figure 6. Separation of 3D HMM log scores for several closely related fold families

(a) NAD(P)-binding domains (red) and FAD-binding domains (blue) scored against FAD 3D HMM.

(b) Thioredoxin (red) and Flavodoxin (blue) domains scored against Thioredoxin 3D HMM.

(c) Lysozyme (red) and Ferrodoxin (blue) domains scored against Lysozyme 3D HMM.

Figure 7. Separation of scores for Thioredoxin (red) and all other less than 95% identical SCOP domains (blue) scored against Thioredoxin 3D HMM.

References

1. Krogh, A., et al., *Hidden Markov models in computational biology. Applications to protein modeling.* J Mol Biol, 1994. **235**(5): p. 1501-31.
2. Rabiner, L.R., *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition.* Proceedings of the IEEE, 1989. **77**(2): p. 257-285.
3. Gribskov, M., R. Lüthy, and D. Eisenberg, *Profile Analysis.* Meth. Enz., 1990. **183**: p. 146-159.
4. Teichmann, S., J. Park, and C. Chothia, *Structural assignments to the proteins of Mycoplasma genitalium show that they have been formed by extensive gene duplications and domain rearrangements.* Proc. Natl. Acad. Sci., 1998. **95**: p. 14658-63.
5. Reese, M.G., et al., *Genie--gene finding in Drosophila melanogaster [see comments].* Genome Res, 2000. **10**(4): p. 529-38.
6. Eddy, S.R., *Hidden Markov models.* Curr. Opin. Struc. Biol., 1996. **6**: p. 361-365.

7. Durbin, R., Eddy, S., Krogh, A., Mitchison, G., *Biological sequence analysis : probalistic models of proteins and nucleic acids*. 1998, Cambridge, UK New York: Cambridge University Press. 356.
8. Bateman, A., Birney, E., Durbin, R., Eddy, S. R., Howe, K. L., Sonnhammer, E. L., *The Pfam protein families database.* Nucleic Acids Research, 2000. **28**(1): p. 263-666.
9. Bystroff, C., Thorsson, V., Baker, D., *HMMSTR: a Hidden Markov Model for Local Sequence-Structure Correlations in Proteins.* Journal of Molecular Biology, 2000. **301**(1): p. 173-190.
10. Sonnhammer, E.L., von Heijne, G., Krogh, A., *A hidden Markov model for predicting transmembrane helices in protein sequences.* Intelligent Systems in Molecular Biology, 1998. **6**: p. 175-82.
11. Karplus, K., et al., *Predicting protein structure using hidden Markov models.* Proteins, 1997. **Suppl 1**: p. 134-9.
12. Di Francesco, V., J. Garnier, and P.J. Munson, *Protein topology recognition from secondary structure sequences: application of the hidden Markov models to the alpha class proteins.* J Mol Biol, 1997. **267**(2): p. 446-63.
13. Di Francesco, V., et al., *Fold recognition using predicted secondary structure sequences and hidden Markov models of protein folds.* Proteins, 1997. **Suppl 1**: p. 123-8.
14. Di Francesco, V., et al., *Incorporating global information into secondary structure prediction with hidden Markov models of protein folds.* Proc Int Conf Intell Syst Mol Biol, 1997. **5**: p. 100-3.
15. Kersting, K., et al., *Towards discovering structural signatures of protein folds based on logical hidden Markov models.* Pac Symp Biocomput, 2003: p. 192-203.
16. Taylor, W.R. and C.A. Orengo, *Protein Structure Alignment.* J. Mol. Biol., 1989. **208**: p. 1-22.
17. Holm, L. and C. Sander, *Protein Structure Comparison by Alignment of Distance Matrices.* J. Mol. Biol., 1993. **233**: p. 123-128.
18. Sali, A. and T.L. Blundell, *The definition of general topological equivalence in protein structures: A procedure involving comparison of properties and relationships through simulated annealing and dynamic programming.* J. Mol. Biol., 1990. **212**: p. 403-428.
19. Godzik, A. and J. Skolnick, *Flexible algorithm for direct multiple alignment of protein structures and sequences.* CABIOS, 1994. **10**: p. 587-596.
20. Artymiuk, P.J., et al., *Searching Techniques for Databases of Protein Structures.* J. Inform. Sci., 1989. **15**: p. 287-298.
21. Gerstein, M. and M. Levitt, *Using Iterative Dynamic Programming to Obtain Accurate Pairwise and Multiple Alignments of Protein Structures*, in *Proc. Fourth Int. Conf. on Intell. Sys. Mol. Biol.* 1996, AAAI Press: Menlo Park, CA. p. 59-67.
22. Gerstein, M. and M. Levitt, *Comprehensive Assessment of Automatic Structural Alignment against a Manual Standard, the Scop Classification of Proteins.* Protein Science, 1998. **7**: p. 445-456.
23. Levitt, M. and M. Gerstein, *A Unified Statistical Framework for Sequence Comparison and Structure Comparison.* Proceedings of the National Academy of Sciences USA, 1998. **95**: p. 5913-5920.
24. Singh, A.P., Brutlag, D. L., *Hierarchical Protein Structure Superposition using both Secondary Structure and Atomic Representations.* Proc. Intelligent Systems for Molecular Biology, 1997. **5**: p. 284-293.

25. Holm, L. and C. Sander, *Structural alignment of globins, phycocyanins and colicin A.* FEBS Lett., 1993. **315**: p. 301-306.
26. Holm, L. and C. Sander, *The FSSP database of structurally aligned protein fold families.* Nuc. Acid Res., 1994. **22**: p. 3600-3609.
27. Bryant, S.H. and S.F. Altschul, *Statistics of sequence-structure threading.* Curr Opin Struct Biol, 1995. **5**(2): p. 236-44.
28. Orengo, C.A., et al., *CATH--a hierarchic classification of protein domain structures.* Structure, 1997. **5**(8): p. 1093-108.
29. Wilson, C.A., J. Kreychman, and M. Gerstein, *Assessing Annotation Transfer for Genomics: Quantifying the Relations between Protein Sequence, Structure and Function through Traditional and Probabilistic Scores.* J Mol Biol, 2000. **297**(1): p. 233-249.
30. Sali, A. and J.P. Overington, *Derivation of rules for comparative protein modeling from a database of protein structure alignments.* Protein Science, 1994. **3**(9): p. 1582-1596.
31. Holm, L. and C. Sander, *Alignment of three-dimensional protein structures: network server for database searching.* Methods Enzymol, 1996. **266**: p. 653-62.
32. Taylor, W.R., T.P. Flores, and C.A. Orengo, *Multiple Protein Structure Alignment.* Prot. Sci., 1994. **3**: p. 2358-2365.
33. Taylor, W.R., *Multiple sequence alignment by a pairwise algorithm.* CABIOS, 1987. **3**: p. 81-87.
34. Taylor, W.R., *A flexible method to align large numbers of biological sequences.* J. Mol. Evol., 1988. **28**: p. 456-474.
35. Taylor, W.R., *Hierarchial method to align large numbers of biological sequences.* Meth. Enz., 1990. **183**: p. 456-473.
36. Gerstein, M. and R. Altman, *Average core structures and variability measures for protein families: Application to the immunoglobulins.* J. Mol. Biol., 1995. **251**: p. 161-175.
37. Altman, R. and M. Gerstein, *Finding an Average Core Structure: Application to the Globins*, in *Proceedings of the Second International Conferene on Intelligent Systems in Molecular Biology*. 1994, AAAI Press: Menlo Park, CA. p. 19-27.

Figure 1: Typical 1D HMM topology (adapted from [7]). Squares, diamonds and circles represent match (M_k), insert (I_k) and delete (D_k) states, respectively. Arrows indicate state-to-state transitions, which may occur according to the corresponding transition probabilities.

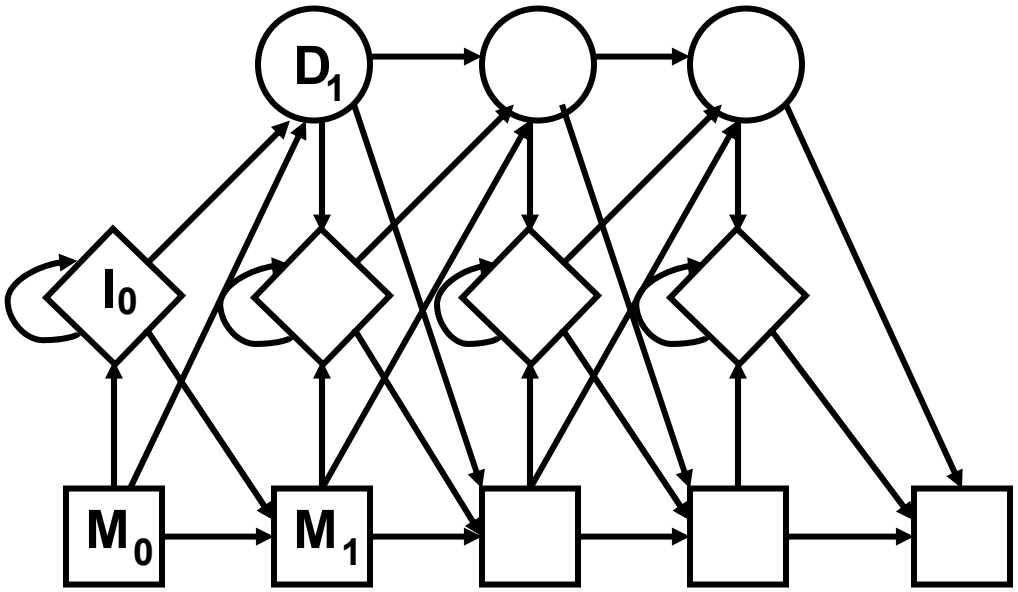

Figure 2: Structural alignment of two protein backbones (PDB ids: 1ECD.pdb and 1HLB.pdb). Aligned parts are shown in yellow.

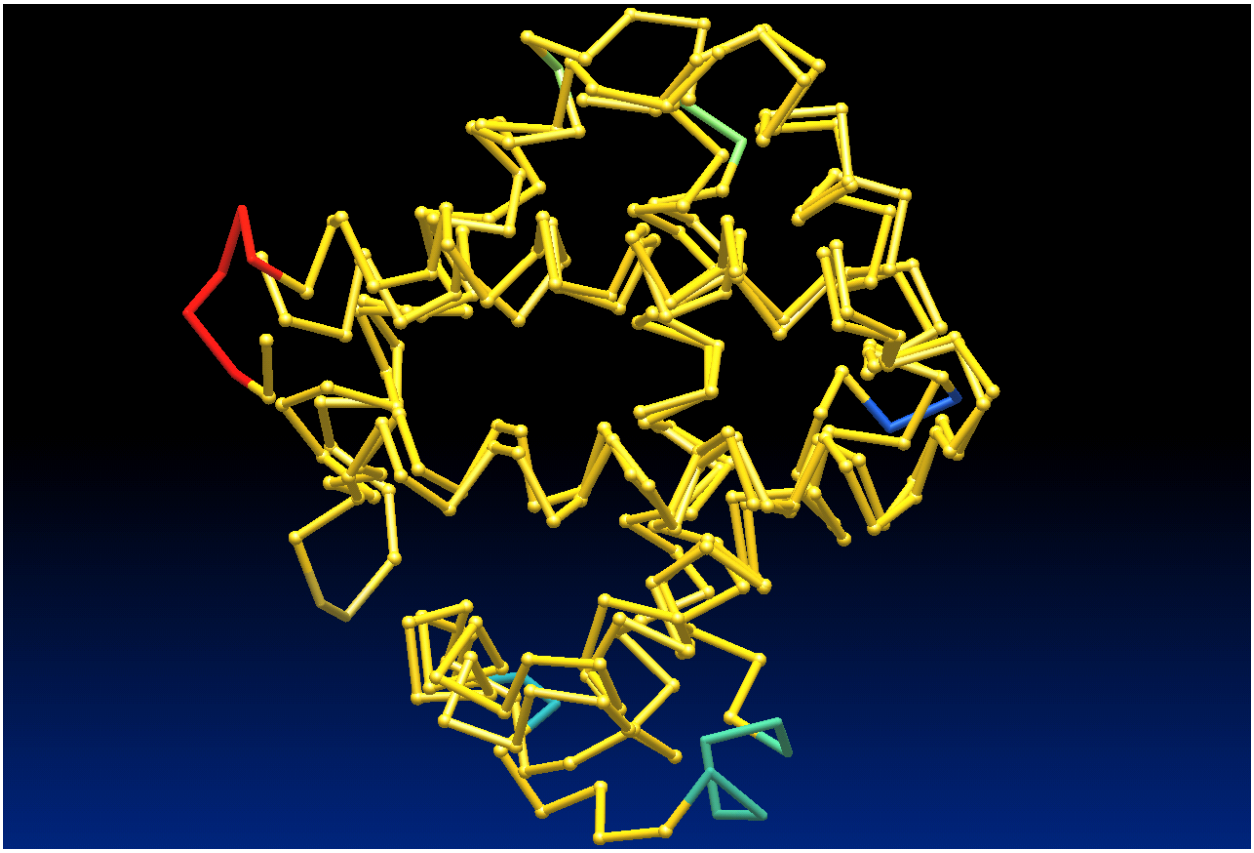

Figure 3: Discretization of the coordinate probability distribution in one dimension.

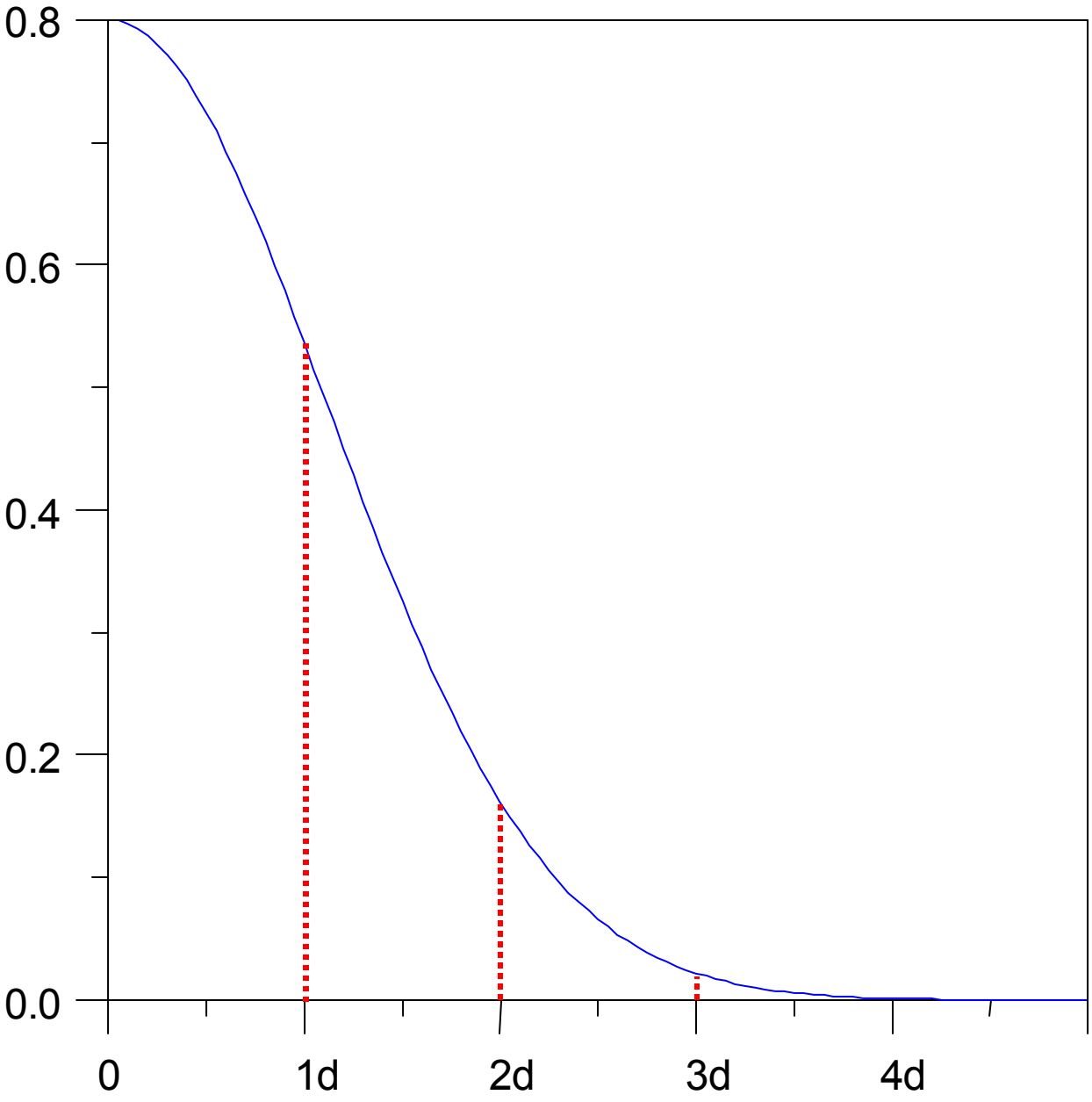

Figure 4. Separation of scores for IgV domains (red) and globin domains (blue) scored against globin 3D HMM.

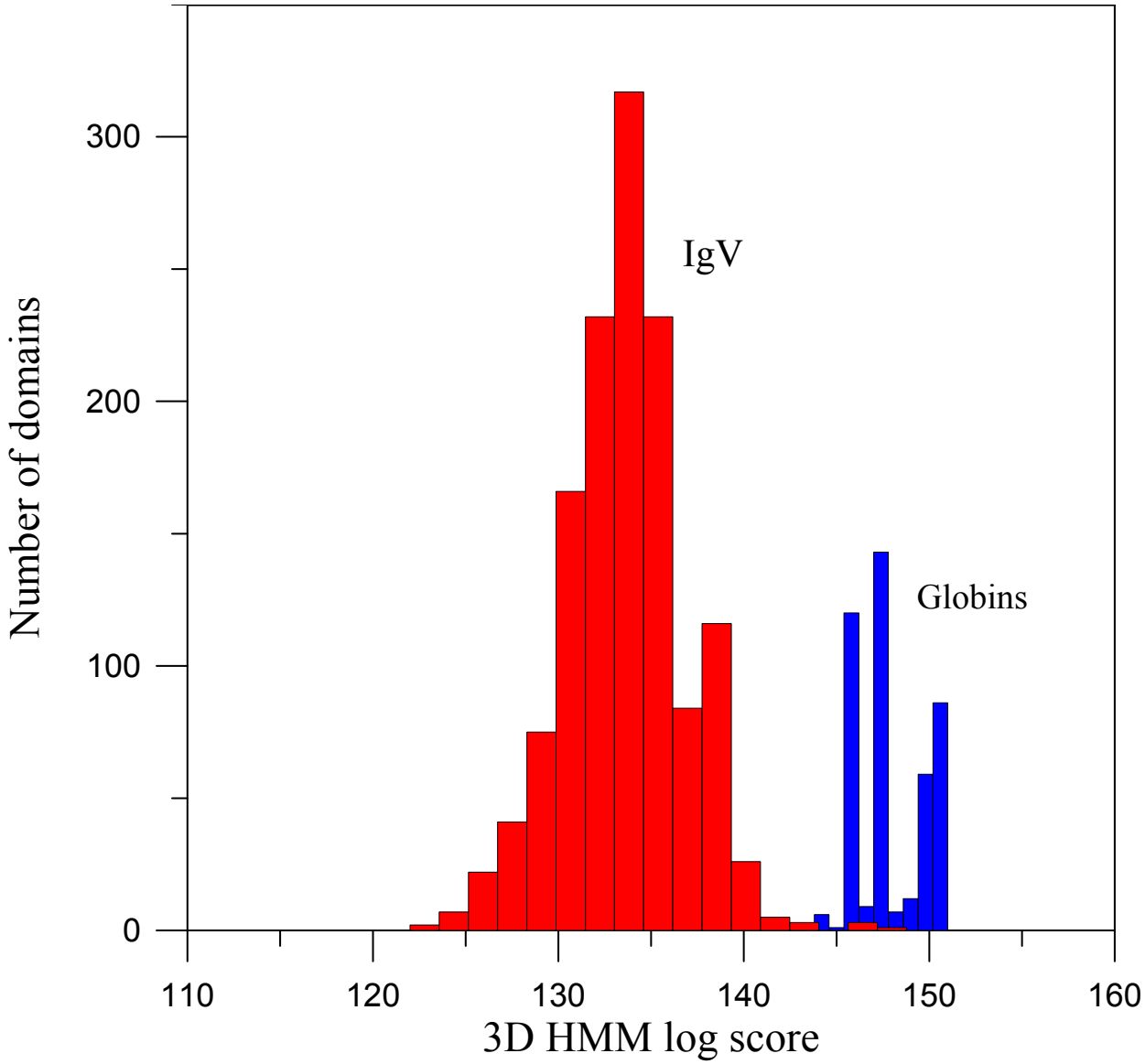

Figure 5. Histogram of RMSD values for globin domains (yellow) and IgV domains (blue) calculated for the alignment of these domains against the IgV core.

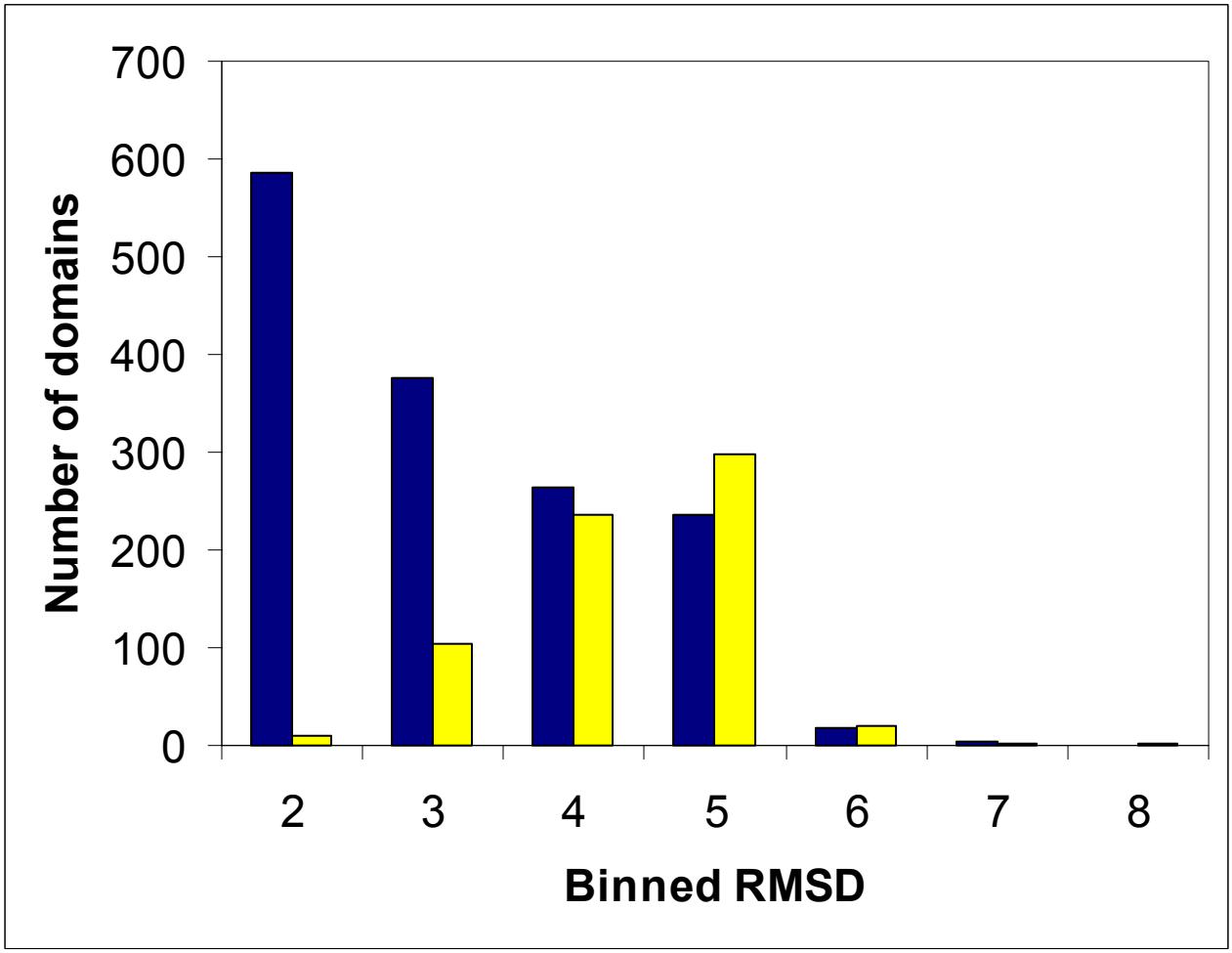

Figure 6a. Separation of scores for NAD(P)-binding domains (red) and FAD-binding domains (blue) scored against FAD 3D HMM.

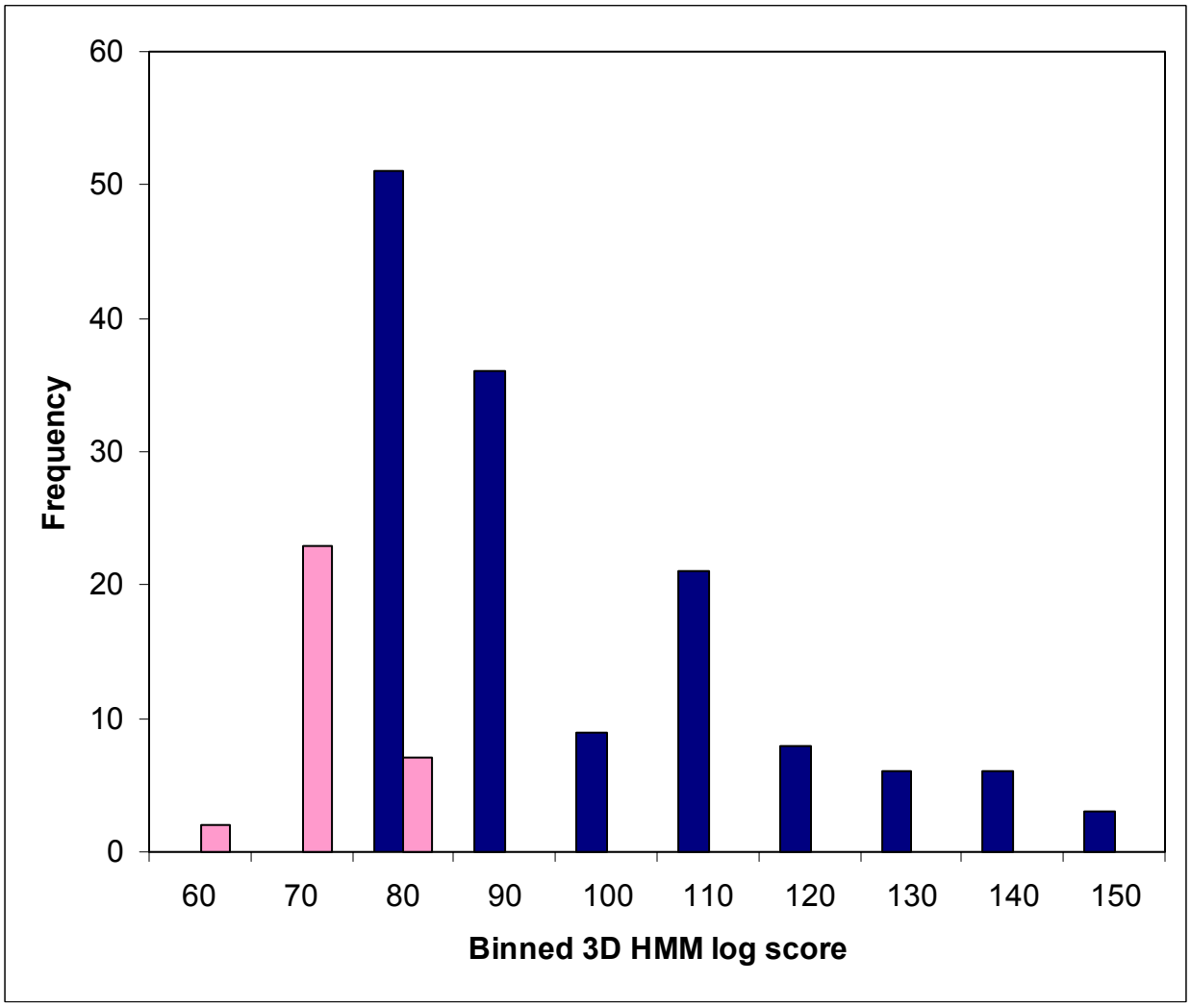

Figure 6b. Separation of scores for Thioredoxin domains (red) and Flavodoxin domains (blue) scored against Thioredoxin 3D HMM.

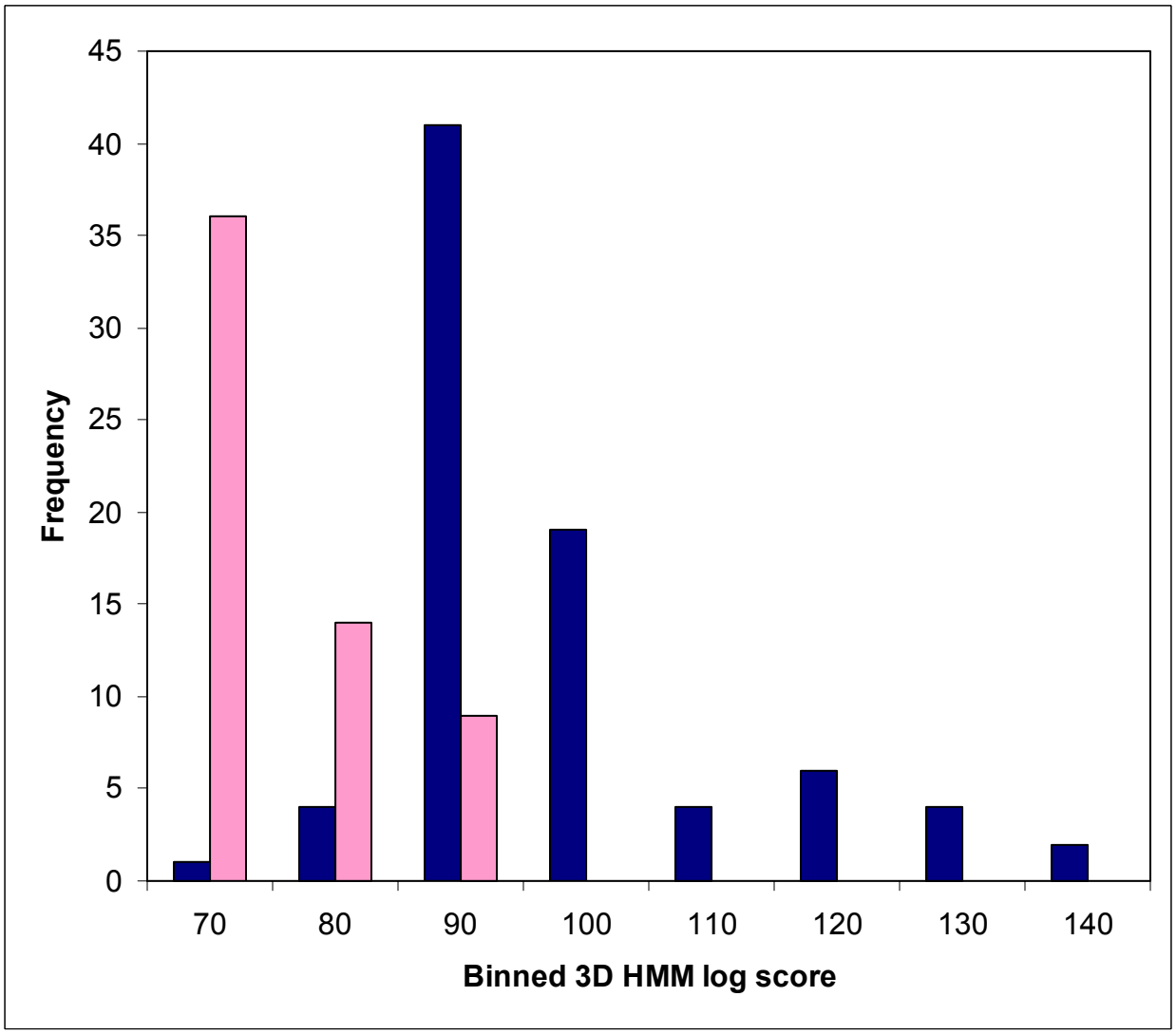

Figure 6c. Separation of scores for Lysozyme domains (red) and Ferrodoxin domains (blue) scored against Lysozyme 3D HMM.

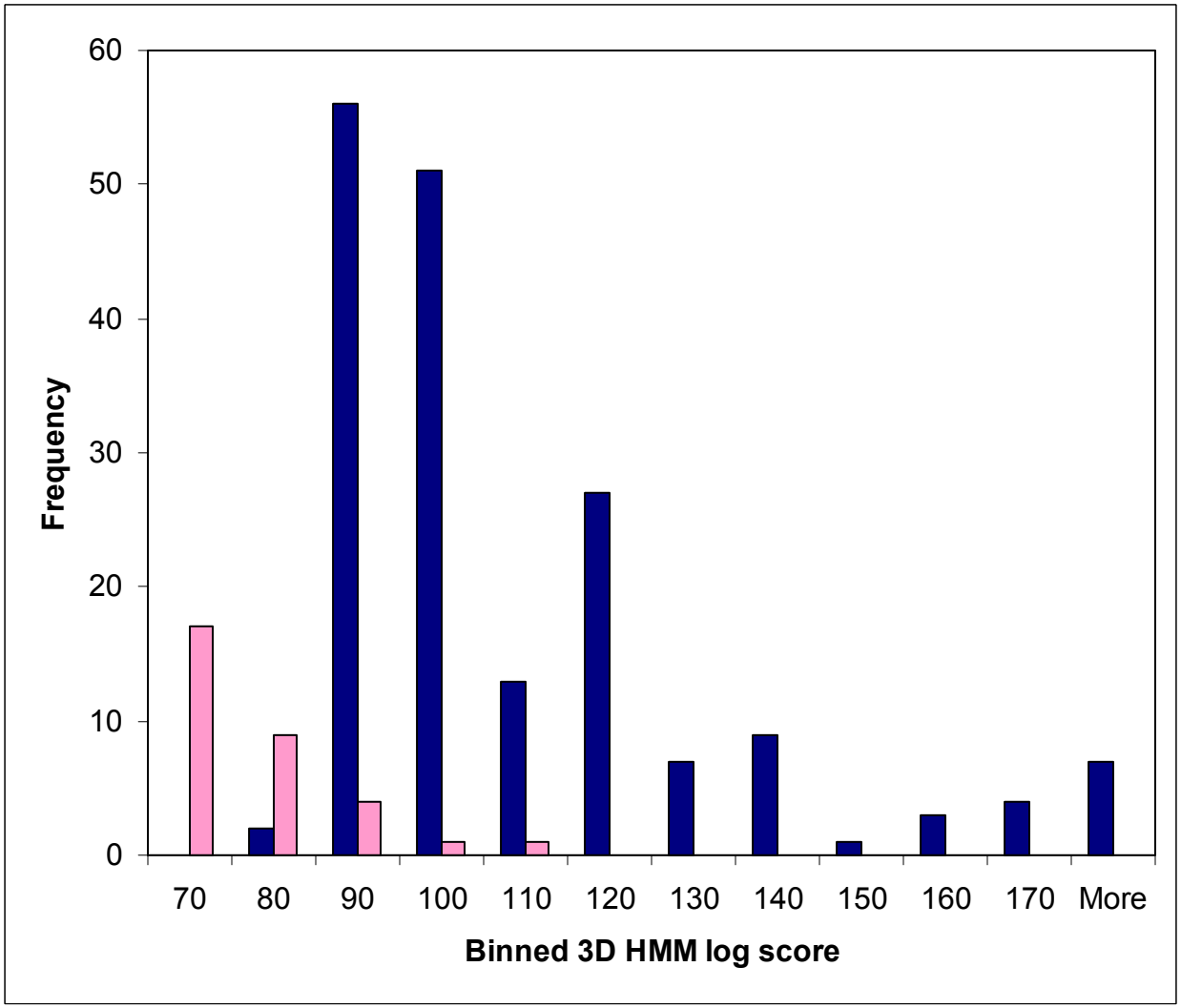

Figure 7. Separation of scores for Thioredoxin domains (red) and all other less than 95% sequence-identical SCOP domains (blue) scored against Thioredoxin 3D HMM.