Touring Protein Space with Matt

Noah M. Daniels, Anoop Kumar, Lenore J. Cowen, and Matt Menke

Abstract—Using the Matt structure alignment program, we take a tour of protein space, producing a hierarchical clustering scheme that divides protein structural domains into clusters based on geometric dissimilarity. While it was known that purely structural, geometric, distance-based measures of structural similarity, such as Dali/FSSP, could largely replicate hand-curated schemes such as SCOP at the family level, it was an open question as to whether any such scheme could approximate SCOP at the more distant superfamily and fold levels. We partially answer this question in the affirmative, by designing a clustering scheme based on Matt that approximately matches SCOP at the superfamily level, and demonstrates qualitative differences in performance between Matt and DaliLite. Implications for the debate over the organization of protein fold space are discussed. Based on our clustering of protein space, we introduce the Mattbench benchmark set, a new collection of structural alignments useful for testing sequence aligners on more distantly homologous proteins.

Index Terms—SCOP, hierarchical classification, structure alignment, fold space, automated classification.

1 Introduction

The accepted gold-standard hierarchical classification systems for protein structural domains, SCOP [23], [2] and CATH [24], [25], [12], have long relied on manual classification methods to organize the hierarchy and place new protein structures within their framework, though CATH always maintained semiautomated methods. Even now, where both SCOP and CATH have switched to hybrid manual/semiautomated methods [12], the automatic methods are still attempting to fit new protein domain folds into an initial classification schema that was derived manually. New modifications to the clustering structure continue to be made by expert biologists based on sequence, evolutionary, and functional information, not solely based on geometric similarity of the placement of atoms in the protein backbone.

On the other hand, pairwise protein structural alignment programs superimpose protein domains to minimize a distance value-based solely on geometric criteria [9]. When such a scheme is coupled with one of many possible methods that create hierarchical clusters based on pairwise distances [31], the result is a fully automatic, unsupervised partitioning of protein structural domains into hierarchical classification systems. Such "bottom-up" protein structure classifications, as they are called in [35], have been previously designed based on VAST [20], [11], Dali [17], [18], [16], and others [39], and have both practical and theoretical appeal. Practically, removing a human expert speeds the assignment of new protein structures to clusters. Theoretically, a mathematical characterization of protein similarity and dissimilarity, if it proves biologically useful or meaningful, is objective, uniformly applied, and gives a human-expert-independent map of the known protein universe.

• *The authors are with the Tufts University, 161 College Avenue, Halligan Hall Room 102, Medford, MA 02155. E-mail: {ndaniels, cowen}@cs.tufts.edu, {anoopkum, mattmenke}@gmail.com.*

Manuscript received 16 Aug. 2010; revised 4 Mar. 2011; accepted 15 Mar. 2011; published online 1 Apr. 2011.
For information on obtaining reprints of this article, please send e-mail to: tcbb@computer.org, and reference IEEECS Log Number TCBB-2010-08-0201. Digital Object Identifier no. 10.1109/TCBB.2011.70.

Unfortunately, it has been found in multiple previous papers that SCOP and CATH hierarchical classifications of protein structure both differ substantially from each other [13], [10], [8] and also from the classification schema that result from automatic bottom-up unsupervised clusterings of protein space [9], [13], [32], [8], [30], even when protein chains are broken up into the more modular unit of "protein domain," as is now done by SCOP, CATH, and most automated schemes [18], [35]. Previous papers have characterized those protein domain clusters on which SCOP and CATH agree [13], [10], [8]. Previous automatic methods seem to be able to be made to match the closest-homology *family* level of the SCOP hierarchy, but were found to diverge considerably at the more distantly homologous *superfamily* and at the quite remotely homologous *fold* levels of the SCOP hierarchy [9], [13], [32], [19], [30], [34], with similar divergence from CATH [13], [14], [8]. This is unfortunate, because, for example, the superfamily level of the SCOP hierarchy clusters proteins that share similar topologies and are believed to have evolved from a common ancestor [23], allowing important inferences to be made about function [30], [35]. Thus, the superfamily level of the SCOP hierarchy has strong biological utility (we focus on SCOP rather than CATH for the remainder of this paper; similar statements can be made about CATH): if a fully automated "bottom-up" distance-based clustering methods cannot approximately replicate it, it is not clearly meaningful or useful.

This ties into a spirited debate among the computational proteins community, about the central question of whether "protein fold space" is *discrete* or *continuous* [28]. A continuous view comes from the theory that modern protein evolved by aggregating fragments of ancient proteins [28], [14], [35], [29]. A discrete view comes from evolutionary process constrained by thermodynamic stability of the structure [29]. In particular, if most mutations move the conformation of a stable folded chain away from an "island" of thermodynamic structural stability, then stabilizing selection will promote fold conservation, and movements between folds will be uncommon [6]. If

geometric distance and evolutionary relation approximately coincide, then an automatic method that approximately matches SCOP at the superfamily level is conceivable.

In this paper, we present a bottom-up automatic hierarchical classification scheme for protein structural domains based on the multiple structure alignment program Matt [21]. Matt, which stands for "multiple alignment with translations and twists" was specifically developed by our group to geometrically align more distantly homologous protein domains. It accomplishes this by allowing flexibility in the form of small, geometrically impossible bends and breaks in a protein structure, in order to distort it into alignment with another protein structure. Matt was shown to perform particularly well compared to competing multiple and pairwise structure alignment programs on proteins whose homology was similar to the SCOP superfamily level [21], [27], [3]. Surprisingly, we find that our automatic classification scheme based on a pairwise distance value derived from Matt, coupled with a straightforward neighbor-joining algorithm to construct the hierarchical clusters [33] matches SCOP better than previous automatic methods, at the superfamily, and even, to some extent, at the fold level. In comparison, the same hierarchical clustering method using a pairwise distance value based on DaliLite [16], a recent implementation of the Dali structural alignment program, replicates previous findings and cannot mimic SCOP on the superfamily level of the SCOP hierarchy. We, thus, conclude that perhaps the threshold at which protein domain space is naturally discrete extends at least through the superfamily level, and that perhaps the manually curated SCOP hierarchy has *geometric* coherence at the superfamily level (and in some parts of the fold hierarchy, see Section 4) so these clusters are intrinsic properties of the geometry of fold space, not just human-generated categories.

A practical implication of our results may be that automatic methods with a Matt-based distance value may ultimately help speed the assignment of new protein structural domains to the appropriate place in the SCOP hierarchy. We note, however, that in fact determining where to place a new structure into an existing hierarchy is a much simpler problem (analogous to "supervised learning") than creating an entire cluster hierarchy from an automatic pairwise distances from scratch (analogous to "unsupervised learning"), and fairly successful methods already exist to correctly place a new structure into the existing SCOP hierarchy [10], [4], [5]. Thus, the primary interest in this result may be that if a Matt distance value can "recover" SCOP superfamilies to a great extent, this validates both automatic and hand-curated methods of classification, and the entire concept of "superfamily" at the same time. Namely, at this level of structural similarity, it appears we may not often have to choose between evolutionary and geometric criteria for structural domain similarity.

A byproduct of our organization of protein space is that by looking at where agreement of our Matt clusters with SCOP is exact, we can construct a new set of gold-standard protein multiple structure alignments of distantly homologous proteins (and associated decoy sets) for which we can have confidence that the Matt structural alignment is meaningful. Thus, we introduce "Mattbench," a set of structural alignments at two levels: superfamilies (consisting of 225 alignments with between 3 and 15 proteins in each alignment set), and folds (consisting of 34 alignments with between 3 and 15 proteins in each alignment set. Mattbench is meant as an alternative to the SABmark [36] benchmark set, which also attempts to mimic SCOP, but Mattbench's alignment sets only cover those subsets of SCOP superfamilies and folds where Matt finds geometric consistency. Thus, while Mattbench is slightly less complete than SABmark in coverage, its alignments are likely to be more consistent, making it a better benchmark on which to test sequence alignment methods. Complete details on how Mattbench is constructed appear in Section 2.6; Mattbench itself can be downloaded from http://www.bcb.tufts.edu/mattbench.

Finally, we remark that this work, like most recent work that compares different hierarchical classification systems, already presumes the "structural domain" as the basic structural unit (as do SCOP and CATH), where many protein structures contain multiple structural domains [18]. The problem of partitioning a protein into its structural domains is far from trivial [37], [15], but there has been much recent progress in computational methods that split a protein structure automatically into domains and find the domain boundaries [15], [26]. In any case, that is not the focus of our current paper, and we assume that the protein has already been correctly split into domains as a preprocessing step.

An extended abstract of this paper appeared in [7].

2 Methods

2.1 Representative Proteins

From the 110,776 protein domains of known structure from ASTRAL version 1.75, we construct a set of representative protein domains filtered to 80 percent identity (according to BLASTP [1]) and a minimum sequence length of 40 residues. This provides a reasonable first pass for identifying groups of similar protein domains, and allows us to shrink the search space significantly. The set of clusters was constructed by running a greedy agglomerative minimum-linkage clustering algorithm based on this threshold of 80 percent sequence identity. This produced 10,418 groups of proteins that shared significant sequence identity.

From each cluster, we identified a representative. First, we discard engineered or mutant proteins, and any proteins whose X-ray crystallography resolution is >5.0 Å, from any cluster that has alternative representatives that meet our criteria. Next, treating each cluster as a (potentially, but not necessarily complete) graph whose nodes are the constituent proteins and whose edge weights are the sequence identity values from the BLASTP alignments with at least 80 percent identity, we consider the weighted degree (sum of edge weights) of each protein, and we favor the proteins with greatest weighted degree. We break ties first by the date the structure was determined (preferring more recent structures), then by the quality of the solved structure. The remaining ties typically come from sequences with $\geq 99\%$ identity, and we break them arbitrarily. The resulting set has 10,418 representative protein domains.

2.2 Distance Values

For these 10,418 representatives, we performed an all-pairs structural alignment using both DaliLite [16], the structural aligner used in the FSSP classification scheme [18] and Matt. In each case, a distance (or dissimilarity) measure is derived for each pair. For DaliLite, the Z-score proved to be a good measure, so we used it without further modification.

For Matt, we used a new distance value that is a modification of the p-value score computed in [21]. Let c be the length of the aligned core shared between the two proteins (in residues), r be the root mean square deviation (RMSD) of the alignment, l_1 and l_2 be the lengths of the two protein domains being aligned (in residues), and k_1, k_2, and k_3 be the constants from the Matt p-value. We compute the distance between two Matt-aligned proteins as follows:

$$d = \frac{1}{k_1 \times (r - k_2 \times \frac{c^2}{\frac{l_1+l_2}{2}} + k_3)}.$$

This value differs from the formula that Matt uses to compute a p-value only in that it squares the core-length term to better weight longer aligned cores (c^2 instead of c). We found this improved performance.

2.3 Distance Threshold

Based on each of the Dali Z-score and Matt distances, we next learned the distance cutoffs that most closely mimicked the family, superfamily, and fold levels of the SCOP hierarchy as follows:

1. Initialize a training set T and a set of already-chosen pairs A.
2. 10,000 times, do:
   a. Choose proteins p and q such that $p \neq q$ and p and q are in the same SCOP grouping, and the pair $p, q \notin A$.
   b. Choose proteins r and s such that $r \neq s$ and r and s are in different SCOP groupings, and the pair $r, s \notin A$.
   c. Add p, q and r, s to A.
   d. Determine the DaliLite or Matt distance between p and q. Call this $d_{p,q}$.
   e. Add $d_{p,q}$ to the training set T with label *true*.
   f. Determine the DaliLite or Matt distance between r and s. Call this $d_{r,s}$.
   g. Add $d_{r,s}$ to the training set T with label *false*.
3. Compute true positive rate R_{tp}, true negative rate R_{tn}, positive rate R_p, and negative rate R_n for T based on the class labels $true$ and $false$.
4. Determine the value of $d_{p,q}$ that results in maximizing the accuracy $\frac{R_{tp}+R_{tn}}{R_p+R_n}$.

In other words, we set $d_{p,q}$ to be the value corresponding to the point on the Receiver Operating Characteristic (ROC) curve that intersects the tangent isoperformance line [38], maximizing the sum $R_{tp} + R_{tn}$. The area under the ROC curve measure (AUC) is a summary statistic that captures how well the pairwise distance score can discriminate between structures that share or do not share SCOP cluster membership.

We note that setting the pairwise distance cutoffs (determining the value of $d_{p,q}$ in Step 4) is the only "supervision" our algorithm uses in constructing its clustering (see discussion below). *Once the three single scalar pairwise distance cutoff (corresponding to SCOP "family," "superfamily," and "fold" levels of dissimilarity) are set, no further information from SCOP is utilized to produce the clustering.*

2.4 Clustering and Tree-Cutting

Based on the distance functions, we computed values for all pairwise alignments based on the Matt or DaliLite output, and represented this as a distance matrix. We ran the ClearCut program [33] in strict neighbor-joining mode ($-$N option) to produce a dendrogram based on these Matt or DaliLite distance values. We then recursively descended this tree to produce family, superfamily, and fold-level groupings as follows: for a given subtree, if all leaves (protein domains) in that subtree are within a threshold t of one another (where t is the family, superfamily, or fold threshold), then those leaves are all merged into a new grouping of that level. Otherwise, we recursively descend into the two subtrees of that subtree's root until we reach a subtree all of whose leaves fall within a given threshold (family, superfamily, or fold; based on Matt distance or DaliLite Z-score as appropriate) of one another. Thus, we are performing a total-linkage clustering, but using the topology of the dendrogram to determine which protein domains get left out of a given cluster.

We remark that Sam et al. [31] did an extensive study of clustering and tree-cutting methods, and looked at their effect on performance for several distance values. They tested three "SCOP-dependent" and seven "SCOP-independent" tree-cutting strategies. However, their "SCOP-independent" strategies all required as input the target number of SCOP clusters to produce at each level. In contrast, our method discovers the number of clusters as an organic function of the protein domain space, based only on a globally learned dissimilarity cutoff; it is, thus, of independent interest that we nearly replicate the number of SCOP clusters at each level (see Table 2).

2.5 Jaccard Similarity Metric

The Jaccard index, or Jaccard similarity coefficient, of two sets A and B is defined as $J(A, B) = \frac{|A \cap B|}{|A \cup B|}$. Based on the Jaccard index of a cluster (e.g., family or superfamily or fold) produced by our algorithm (a Matt family or DaliLite family) and a SCOP grouping of the same level, and looking at the identity of protein domains in the two groupings, we can compare how alike they are. We can, thus, easily find the most similar SCOP family to each Matt family, $S \rightarrow M$ and vice versa, $M \rightarrow S$. This directional mapping is neither one-to-one nor onto, but each cluster on the "source" side will be mapped to some most similar cluster on the "sink" side. The resulting directed graph allows us to explore the distribution of Jaccard indices as well as the distribution of degrees of each cluster. A perfect matching would correspond to every Jaccard index being 1.0, and every cluster having degree 1. Clearly, we do not expect to achieve a perfect matching, but this metric allows us to compare the quality of clustering, relative to SCOP, of our algorithm using the Matt distance and the DaliLite Z-score distance.

Each direction of the metric is produced as follows, using as an example the comparison of Matt families to SCOP families. Consider the set of Matt families and SCOP

families as a bipartite graph, with the Matt families on one side of the bipartition and the SCOP families on the other. Initially, the graph has no edges. For each Matt family, find the most similar (by Jaccard index) SCOP family. A weighted, directed edge is drawn from each Matt family to its most similar SCOP family; the edge weight is equal to the Jaccard index, which ranges from 0 to 1. This is performed until each Matt family has been matched to a SCOP family. This process is repeated in the other direction, matching each SCOP family to its most similar Matt family, and the same thing is done for Matt and DaliLite at the superfamily and fold levels of the SCOP hierarchy.

Recall that in this analysis, as is standard [13], we are considering only the protein domains that were identified as cluster representatives within each group of protein domains that share 80 percent sequence identity.

2.6 Benchmark Set

With the hierarchy of Matt-derived folds, superfamilies, and families constructed, we produced a benchmark set of protein alignments at two levels: superfamilies (consisting of 225 alignments), and folds (also referred to as the "twilight zone" of protein homology, consisting of 34 alignments). At the superfamily level, we generated the benchmark set as follows:

1. Choose Matt superfamilies that contain at least three representative proteins.
2. For each Matt superfamily:
   a. Identify the most similar SCOP superfamily (by Jaccard index) and take the intersection of it and the Matt superfamily. Call this set S.
   b. Run BLAST on all pairs of proteins in S, storing the maximum e-value as E.
   c. For any pair of proteins $p, q \in S$ that share greater than 50 percent sequence identity, remove the shorter one (breaking ties arbitrarily by alphabetic order of protein name). Call this set S'. Proceed if and only if S' still has at least three proteins.
   d. Run a Matt multiple alignment on S', and store this alignment as the Mattbench alignment for S'.
3. For each Mattbench superfamily S, produce a decoy set D as follows:
   a. Consider every Matt representative protein $p \notin S$. For each p:
      i. Discard p if it is in the most similar (by Jaccard index) SCOP superfamily to p's Matt superfamily.
      ii. Run BLAST on p against every protein $s \in S$, storing the e-value $e_{s,p}$ and sequence identity $i_{s,p}$.
      iii. Run Matt on p against every protein $s \in S$, storing the Matt distance $m_{s,p}$.
      iv. Discard p if $\exists s$ such that $i_{s,p} \geq 50\%$.
      v. Discard p unless $\exists s$ such that $e_{s,p} < E$ (this is the E stored as the maximum e-value above).
      vi. Discard p unless $\forall s, m_{s,p} > T_{superfamily}$ (the superfamily threshold used in Matt clustering).
      vii. If p has not been discarded, add it to the benchmark decoy set D.

The "twilight zone" benchmark set is generated in an identical manner, except that the Matt and SCOP fold levels are used, and the sequence identity cutoff is 20 percent rather than 50 percent. The BLAST E-value criterion is the same used by SABmark [36] and makes sure each decoy is a useful decoy rather than an obvious negative match. The Matt distance criterion is present because if the decoy protein is within the threshold of some protein in that superfamily, the decoy is only *not* in that superfamily because of the overall topology of the cluster. Both benchmarks can be found at http://www.bcb.tufts.edu/mattbench.

TABLE 1
ROC Area for Pairwise Performance versus SCOP

	Matt	DaliLite
Families	0.922	0.958
Superfamilies	0.842	0.615
Folds	0.840	0.871

While DaliLite slightly outperforms Matt at family and fold levels, Matt significantly outperforms DaliLite at the superfamily level.

3 Results

3.1 Pairwise Distance Comparisons

We first asked if a pairwise Matt or DaliLite distance cutoff could correctly distinguish among pairs of proteins that were in the same SCOP cluster from those that were not. Table 1 shows the AUC at the SCOP family, superfamily, and fold level, for the Matt and DaliLite distance scores. Note that at the family and fold levels, these values are very close (DaliLite outperforms Matt by a small margin), but at the superfamily level, Matt significantly outperforms DaliLite, achieving 0.842 ROC Area versus DaliLite's 0.615. Matt was developed to better align structures at the superfamily level of homology, but the size of the gap in ROC AUC is still surprising. We further remark that at the fold level, DaliLite's seemingly competitive performance is somewhat illusory, since it is shattering many SCOP folds, each into many tiny pieces (see below).

3.2 Clustering Performance

While the pairwise performance of Matt compared to DaliLite at the superfamily level is impressive, pairwise similarity does not necessarily translate into better clustering performance. Thus, it is Matt's clustering performance we explore next. First, we give the simplest possible comparison; raw numbers of clusters produced by Matt and DaliLite compared to SCOP at the three levels. Recall that unlike the clustering algorithm explored in [31], the number of clusters produced by our dendrogram and tree-cutting method is a direct consequence of the pairwise distance threshold, and is not artificially set to match SCOP (see Section 2.4). Table 2 shows that the Matt clustering produces approximately the same number of clusters as

TABLE 2
Number of Clusters at Each Level for Each Method

	SCOP	Matt	DaliLite
Families	3471	3498	3081
Superfamilies	1656	1716	2455
Folds	981	891	2277

Matt more closely matches the number of families, superfamilies, and folds in SCOP than DaliLite does. DaliLite clustering results in too few families, but too many superfamilies and folds with respect to SCOP.

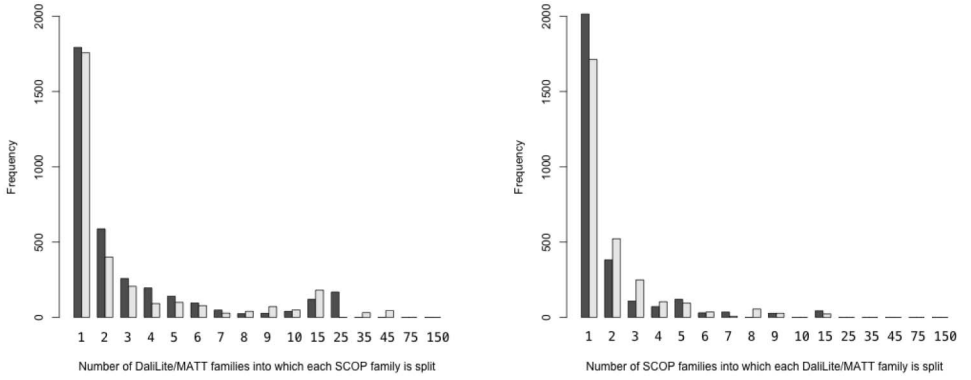
SCOP at all three levels. While DaliLite also produces approximately the same number of clusters at the family level, at the superfamily and fold levels it produces many more clusters than SCOP. We explore exactly how both methods split and merge SCOP clusters in more detail next.

The Jaccard index serves as a good indicator of how well Matt and DaliLite match SCOP. As the raw numbers of clusters in Table 2 suggest, DaliLite often shatters SCOP superfamilies into multiple clusters. DaliLite also shatters SCOP folds into many more shards on average than Matt. How can this be given the very similar pairwise classification performance at the fold level? We defer this question until the discussion section. We note that even at the family level, Matt performs slightly better than DaliLite at both the average degree and average Jaccard similarity metrics, as shown in Table 3. The average number of Matt or DaliLite families that match to a single SCOP family is between 3.5 and 4; however, notice that a large majority of Matt or DaliLite families map to a single SCOP family and the average is pulled up by a few outliers (see histograms in Fig. 2). Average degree values at the superfamily and fold levels stay nearly constant for Matt, whereas DaliLite's average degree values rise to 16.61 for the superfamily level and 26.57 at the fold level. In the other direction, considering how many Matt or DaliLite clusters span multiple SCOP clusters, at the family level the average degree for Matt and DaliLite are nearly identical (between 1.8 and 2). At the superfamily and fold levels, we would expect DaliLite to outperform Matt by virtue of the fact that it creates many smaller clusters (see Table 2), and DaliLite does, but by a fairly small margin (1.4-1.7 at the superfamily level and 1.1-2 at the fold level). The distributions are displayed in more detail in the histograms in Figs. 1, 2, and 3.

TABLE 3
Descriptive Statistics for the Family, Superfamily, and Fold Levels of Classification

Family	Max Deg.	μ Deg.	σ Deg.	Min Sim.	μ Sim.	σ Sim.
Matt →SCOP	30	3.63	5.470	0.005	0.611	0.373
DaliLite → SCOP	45	3.902	6.919	0.001	0.598	0.380
SCOP → Matt	15	1.873	2.160	0.127	0.712	0.336
SCOP → DaliLite	12	1.983	1.823	0.001	0.655	0.347
Superfamily	**Max Deg.**	**μ Deg.**	**σ Deg.**	**Min Sim.**	**μ Sim.**	**σ Sim.**
Matt → SCOP	28	3.633	5.094	0.003	0.587	0.389
DaliLite → SCOP	153	16.61	36.54	0.001	0.428	0.406
SCOP → Matt	15	1.704	1.913	0.020	0.714	0.326
SCOP → DaliLite	10	1.470	1.229	0.001	0.713	0.324
Fold	**Max Deg.**	**μ Deg.**	**σ Deg.**	**Min Sim.**	**μ Sim.**	**σ Sim.**
Matt → SCOP	18	3.719	4.258	0.004	0.467	0.354
DaliLite → SCOP	149	26.57	40.87	0.001	0.321	0.389
SCOP → Matt	6	1.958	1.122	0.022	0.512	0.326
SCOP → DaliLite	3	1.117	0.353	0.001	0.758	0.299

μ degree is the average number of clusters from the first scheme that map to a single cluster in the second, and σ degree gives the standard deviation. Similarly, we give min, μ, and σ of the Jaccard similarity.



(a) Number of Matt vs. DaliLite families into which each SCOP family is shattered

(b) Number of SCOP families into which each Matt or DaliLite family is shattered

Fig. 1. Family level splitting behavior.

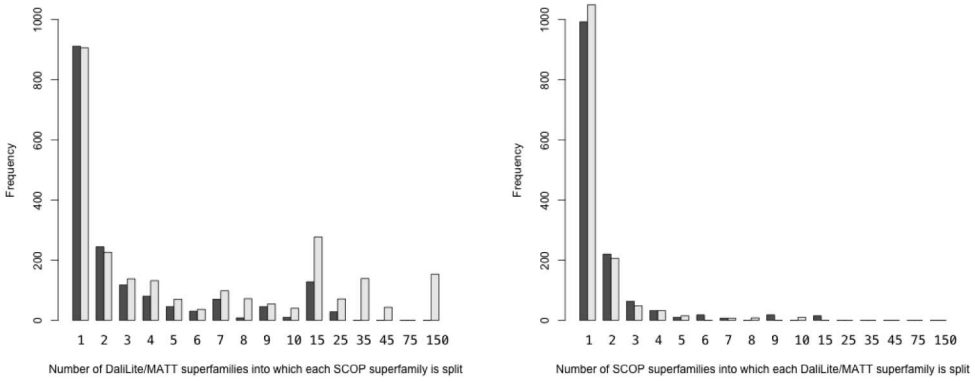
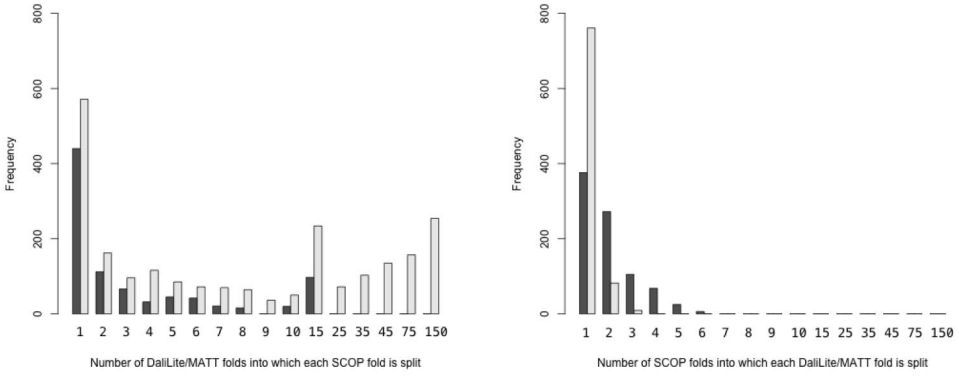


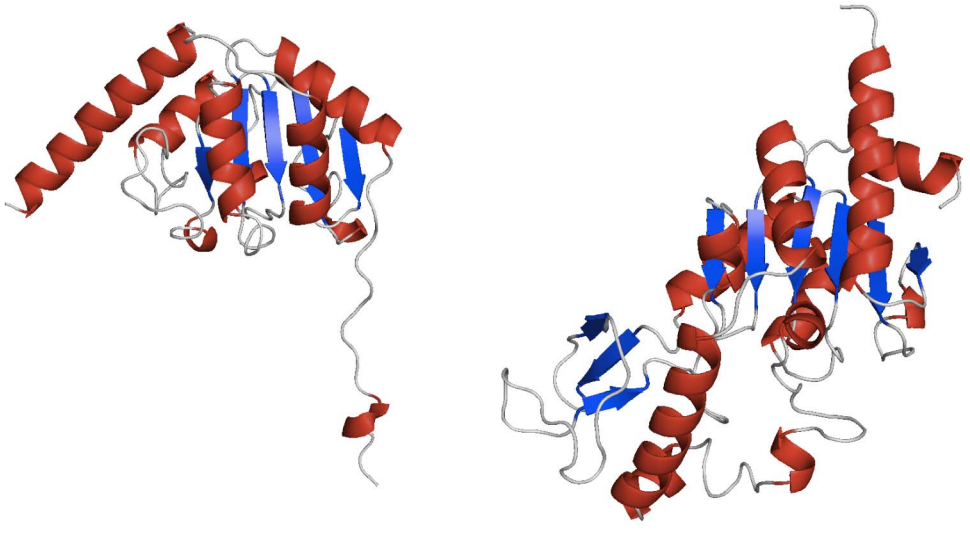
Fig. 2. Superfamily level splitting behavior. (a) Number of Matt versus DaliLite superfamilies into which each SCOP superfamily is shattered. (b) Number of SCOP superfamilies into which each Matt or DaliLite superfamily is shattered.



(a) Number of Matt vs. DaliLite folds into which each SCOP fold is shattered

(b) Number of SCOP folds into which each Matt or DaliLite fold is shattered

Fig. 3. Fold level splitting behavior.



(a) **Example from Matt superfamily 252** Pyruvate oxidase from *Lactobacillus plantarum*, PDB ID 2ez9:a

(b) **Example from Matt superfamily 212** AF1676, Sir2 homolog (Sir2-AF1) from Archaeon *Archaeoglobus fulgidus* , PDB ID 1m2k:a

Fig. 4. Example of a SCOP superfamily split by Matt.

3.3 Specific Example

We thought it would be illuminating to provide a pictorial example of a single SCOP superfamily that Matt splits into two superfamilies. Consider the SCOP superfamily "DHS-like NAD/FAD-binding domain" (c.31.1). There are 24 proteins from this superfamily in our representative set. Matt places 17 of them in one superfamily, but the remaining 7 in a different superfamily. Fig. 4a gives an example protein from the Matt superfamily of size 17, while Fig. 4b gives an example protein from the Matt superfamily of size 7. Both Matt superfamilies contain the same single flat β-sheet of six or seven strands, surrounded by α-helices. In addition,

the proteins in the Matt superfamily of size 7 have a second short 3-4 strands β-sheet. The second short β-sheet is physically on one end of the first β-sheet in 3D space, but sometimes occurs between the second to last and last β-strands in the first β-sheet in terms of linear (sequence) ordering, or else at the very end. The second β-strand is also partially surrounded by α-helices.

Because of the common central motif, it is very possible that these proteins are evolutionarily related and thus belong in the same SCOP superfamily. However, geometrically, the additional short β-sheet is significant enough for Matt to place them in different superfamilies. Matt does, however, place them in the same fold.

4 Discussion

We have shown that using more modern structure alignment programs, an automatic clustering method that approximates SCOP at a superfamily level may be feasible. Of course, any mapping between clusters based on geometric equivalence, and clusters seeking to capture evolutionary and geometric equivalence using information beyond geometry will be imperfect—yet the Matt clusters at the superfamily level seem sufficiently interesting that differences between Matt and SCOP could be illuminating.

As noted earlier, DaliLite tends to shatter SCOP folds into many more shards than Matt. How can this be given the very similar pairwise classification performance at this level? One possibility is that the Matt-based distance value is more stable in regions far beyond the specific thresholds we learned, and that this leads to the topology of the resulting dendrogram (before cutting) more faithfully representing the relationships between more and less closely related folds. In other words, DaliLite's Z-scores may result in more "spoilers" that break up clusters (due to our total-linkage requirement) than Matt's distance value. While we have only compared Matt to DaliLite, comparisons to other aligners such as TM-Align [40] would undoubtedly be interesting.

An interesting question is what Matt clustering results mean for protein fold space at the "fold" level of structural homology. Here, while the Matt clustering clearly seems more informative than that produced by DaliLite, performance is still uneven. There seem to be some SCOP folds where the Matt split appears meaningful, and others where it is more arbitrary. For example, a notoriously difficult SCOP fold for multiple automatic methods is the enormous β/α TIM barrel fold. SCOP places 33 separate superfamilies into this one fold, but both of our clustering approaches seem to split this into multiple folds. For example, DaliLite splits the TIM barrel SCOP fold into 106 separate folds. Matt splits the TIM barrel SCOP fold into "only" 17 separate folds, which is better than 106, but inspection of the boundaries between these Matt fold classes shows more continuity of shape, and the cuts appear to be somewhat arbitrary.

Thus, while touring protein space with Matt seems to lend support to a more discrete view of protein space through the superfamily level, further study of individual clusters may be warranted to determine the breakpoint distance at which continuity takes over. Perhaps the degree of similarity of different individual SCOP folds can be characterized, similarly to what Suhrer et al. [34] did at the family level.

We have made the Mattbench benchmark set available at www.bcb.tufts.edu/mattbench. We hope that developers of protein sequence alignment tools will consider testing their performance on Mattbench whose alignments may be more consistent and reliable than those of the SABmark [36] benchmark, while still being alignments of sequences that are more distant homologous than those of the popular and HOMSTRAD [22] benchmark.

Acknowledgments

This work was funded in part by NIH grant 1R01GM080330-01A1 (to LC).

References

[1] S. Altschul, T. Madden, A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and L. Lipman, "Gapped BLAST and PSI-BLAST: A New Generation of Protein Database Search Programs," *Nucleic Acids Research,* vol. 25, pp. 3389-3402, 1997.

[2] A. Andreeva, D. Howorth, S. Brenner, T. Hubbard, C. Chothia, and A. Murzin, "SCOP Database in 2004: Refinements Integrate Structure and Sequence Family Data," *Nucleic Acids Research,* vol. 32, pp. D226-229, 2004.

[3] C. Berbalk, C. Schwaiger, and P. Lackner, "Accuracy Analysis of Multiple Structure Alignments," *Protein Science,* vol. 18, pp. 2027-2035, 2009.

[4] S. Cheek, Y. Qi, S. Krishna, L. Kinch, and N.V. Grishin, "SCOPmap: Automated Assignment of Protein Structures to Evolutionary Superfamilies," *BMC Bioinformatics,* vol. 7, article 197, 2006.

[5] P.-H. Chi, C.-R. Shyu, and D. Xu, "A Fast SCOP Fold Classification System Using Content-Based E-Predict Algorithm," *BMC Bioinformatics,* vol. 7, article 362, 2006.

[6] I.-G. Choi and S.-H. Kim, "Evolution of Protein Structural Classes and Protein Sequence Families," *Proc. Nat'l Academy of Science USA,* vol. 103, pp. 14056-14061, 2006.

[7] N. Daniels, A. Kumar, L. Cowen, and M. Menke, "Touring Protein Space with Matt," *Proc. Int'l Symp. Bioinformatics Research and Applications,* vol. 6053, pp. 18-28, Jan. 2010.

[8] R. Day, D. Beck, R. Armen, and V. Daggett, "A Consensus View of Fold Space: Combining SCOP, CATH, and the Dali Domain Dictionary," *Protein Science,* vol. 12, pp. 2150-2160, 2003.

[9] M. Gerstein and M. Levitt, "Comprehensive Assessment of Automatic Structural Alignment against a Manual Standard, the SCOP Classification of Proteins," *Proc. Protein Science,* pp. 445-456, 1998.

[10] G. Getz, M. Vendruscolo, D. Sachs, and E. Domany, "Automatic Assignment of SCOP and CATH Protein Structure Classifications from FSSP Scores," *Proteins: Structure Function and Genetics,* vol. 46, pp. 405-415, 2002.

[11] J. Gibrat, T. Madej, and S. Bryant, "Surprising Similarities in Structure Comparison," *Current Opinion in Structural Biology,* vol. 6, pp. 377-385, 2006.

[12] L. Greene, T. Lewis, S. Addou, A. Cuff, T. Dallman, M. Dibley, O. Redfern, F. Pearl, R. Nambudiry, A. Reid, I. Silitoe, C. Yeats, J. Thornton, and C. Orengo, "The CATH Domain Structure Database: New Protocols and Classification Levels Give a More Comprehensive Resource for Exploring Evolution," *Nucleic Acids Research,* vol. 35, pp. D291-297, 2007.

[13] C. Hadley and D. Jones, "A Systematic Comparison of Protein Structure Classifications: SCOP, CATH, and FSSP," *Structure,* vol. 7, pp. 1099-1112, 1999.

[14] A. Harrison, F. Pearl, R. Mott, J. Thornton, and C. Orengo, "Quantifying the Similarity within Fold Space," *J. Molecular Biology,* vol. 323, pp. 909-926, 2002.

[15] T. Holland, S. Veretnik, I.N. Shindyalov, and P. Bourne, "Partitioning Protein Structures into Domains: Why Is It So Difficult?" *J. Molecular Biology,* vol. 361, pp. 562-590, 2006.

[16] L. Holm and J. Park, "DaliLite Workbench for Protein Structure Comparison," *Bioinformatics,* vol. 16, pp. 566-567, 2000.

[17] L. Holm and C. Sander, "Mapping the Protein Universe," *Science,* vol. 260, pp. 595-602, 1996.
[18] L. Holm and C. Sander, "Touring Protein Fold Space with Dali/FSSP," *Nucleic Acids Research,* vol. 26, pp. 316-319, 1998.
[19] R. Kolodny, D. Petrey, and B. Honig, "Protein Structure Comparison: Implications for the Nature of Fold Space, and Structure and Function Prediction," *Current Opinion in Structural Biology,* vol. 16, pp. 393-398, 2006.
[20] T. Madej, J.-F. Gibrat, and S. Bryant, "Threading a Database of Protein Cores," *Proteins,* vol. 23, pp. 356-369, 1995.
[21] M. Menke, B. Berger, and L. Cowen, "Matt: Local Flexibility Aids Protein Multiple Structure Alignment," *PLoS Computational Biology,* vol. 4, no. 1, p. e10, 2008, doi:10.1371/journal.pcbi.0040010.
[22] K. Mizuguchi, C. Deane, T. Blundell, and J. Overington, "HOMSTRAD: A Database of Protein Structure Alignments for Homologous Families," *Protein Science,* vol. 11, pp. 2469-2471, 1998.
[23] 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 Biology,* vol. 297, pp. 536-540, 1995.
[24] C. Orengo, A. Michie, S. Jones, D. Jones, M. Swindells, and J. Thornton, "Cath—A Hierarchic Classification of Protein Domain Structures," *Structure,* vol. 5, no. 8, pp. 1093-1108, 1997.
[25] F. Pearl, C. Bennett, J. Bray, A. Harrison, N. Martin, A. Shepherd, I. Sillitoe, J. Thornton, and C. Orengo, "The CATH Database: An Extended Protein Family Resource for Structural and Functional Genomics," *Nucleic Acids Research,* vol. 31, pp. 452-455, 2003.
[26] O. Redfern, A. Harrison, T. Dallman, F. Pearl, and C. Orengo, "CATHEDRAL: A Fast and Effective Algorithm to Predict Folds and Domain Boundaries from Multidomain Protein Structures," *PLOS Computational Biology,* vol. 3, p. e232, 2007, doi:10.1371/journal.pcji.0030232.
[27] J. Rocha, J. Segura, R. Wilson, and S. Dasgupta, "Flexible Structural Protein Alignment by a Sequence of Local Transformations," *Bioinformatics,* vol. 25, pp. 1625-1631, 2009.
[28] B. Rost, "Did Evolution Leap to Create the Protein Universe?" *Current Opinion in Structural Biology,* vol. 12, pp. 409-416, 2002.
[29] R. Sadreyev, B.-H. Kim, and N. Grishin, "Discrete-Continous Duality of Protein Structure Space," *Current Opinion in Structural Biology,* vol. 19, pp. 321-328, 2009.
[30] V. Sam, C. Tai, J. Garnier, J.F. Gibrat, B. Lee, and P. Munson, "ROC and Confusion Analysis of Structure Comparison Methods Identify the Main Causes of Divergence from Manual Protein Classification," *BMC Bioinformatics,* vol. 7, article 206, 2006.
[31] V. Sam, C. Tai, J. Garnier, J.F. Gibrat, B. Lee, and P. Munson, "Towards an Automatic Classification of Protein Structural Domains Based on Structural Similarity," *BMC Bioinformatics,* vol. 9, article 74, 2008.
[32] I. Shindyalov and P. Bourne, "An Alternative View of Protein Fold Space," *Proteins,* vol. 38, pp. 513-514, 2000.
[33] M. Simonsen, T. Mailund, and C.N.S. Pedersen, "Rapid Neighbour-Joining," *Proc. Eighth Int'l Workshop Algorithms in Bioinformatics (WABI '08),* pp. 113-122, 2008.
[34] S. Suhrer, M. Wederstein, and M. Sippl, "QSCOP-SCOP Quantified by Structural Relationships," *Bioinformatics,* vol. 23, pp. 513-514, 2007.
[35] R. Valas, S. Yang, and P. Bourne, "Nothing about Protein Structure Classification Makes Sense Except in the Light of Evolution," *Current Opinion in Structural Biology,* vol. 19, pp. 392-334, 2009.
[36] I. VanWalle, I. Lasters, and L. Wyns, "SABmark—A Benchmark for Sequence Alignment that Covers the Entire Known Fold Space," *Bioinformatics,* vol. 21, pp. 1267-1268, 2005.
[37] S. Veretnik, P. Bourne, N. Alexandrov, and I. Shindyalov, "Toward Consistent Assignment of Structural Domains in Proteins," *J. Molecular Biology,* vol. 339, pp. 647-678, 2004.
[38] M. Vuk and T. Curk, "Roc Curve, Lift Chart and Calibration Plot," *Metodolo sÿki zvezki,* vol. 2, pp. 89-108, 2006.
[39] A. Zemla, B. Geisbrecht, J. Smith, M. Lam, B. Kirkpatrick, M. Wagner, T. Slezak, and C. Zhou, "STRALCP-Structure Alignment-Based Clustering of Proteins," *Nucleic Acids Research,* vol. 35, p. e150, 2007.
[40] Y. Zhang and J. Skolnick, "TM-Align: A Protein Structure Alignment Algorithm Based on the TM-Score," *Nucleic Acids Research,* vol. 33, no. 7, pp. 2302-2309, 2005.

Noah M. Daniels received the MS degree from Tufts University in 2005. Currently, he is working toward the PhD degree in the Department of Computer Science of Tufts University. His current research interests include protein structural alignment and remote homology detection. He is a student member of the IEEE Computer Society.

Anoop Kumar received the MS degree from Tufts University in 2004, and the PhD degree from Tufts University in 2010. He was a computational biologist at the Broad Institute of MIT and Harvard. He is now a scientist with BBN Technologies.

Lenore J. Cowen received the BA and PhD degrees in mathematics from Yale and MIT, respectively. After finishing her PhD degree in 1993, she was a US National Science Foundation (NSF) postdoctoral fellow and then joined the faculty of the Mathematical Sciences Department at Johns Hopkins University where she was promoted to the rank of an associate professor in 2000. She is a professor in the Computer Science Department at Tufts University. She also has a courtesy appointment in the Tufts Mathematics Department. Lured by the Boston area, and the prospect of making an impact in a growing young department, she joined Tufts in September, 2001. She has been named an ONR Young Investigator and a fellow of the Radcliffe Institute for Advanced Study. She is on the editorial boards of the *SIAM Journal on Discrete Mathematics* and of *SIAM Review*. Her research interests span three areas: discrete mathematics (since high school), algorithms (since 1991 in graduate school), and computational molecular biology (since 2000).

Matt Menke received the PhD degree from MIT in 2009. He worked on the present paper as a postdoctoral fellow at Tufts, and is now at Google Boston. He was the first author on the Matt multiple structure alignment program, and blames his coauthors for the fact that the program shares his name.