Supplementary Information

Non-covalent Lasso Entanglements in Folded Proteins: Prevalence, Functional Implications, and Evolutionary Significance

Viraj Rana[1, ǂ], Ian Sitarik[1, ǂ], Justin Petucci[2], Yang Jiang[1], Hyebin Song[3,4, *],
Edward P. O'Brien[1,2,3, *]

[1] Department of Chemistry, Pennsylvania State University, University Park, Pennsylvania, United States
[2] Institute for Computational and Data Sciences, Pennsylvania State University, University Park, Pennsylvania, United States
[3] Bioinformatics and Genomics Graduate Program, The Huck Institutes of the Life Sciences, Pennsylvania State University, University Park, Pennsylvania, United States
[4] Department of Statistics, Pennsylvania State University, University Park, Pennsylvania, United States

ǂ These authors contributed equally to this research project
* To whom correspondence should be addressed: epo2@psu.edu and hps5320@psu.edu.

Clustering Algorithm [ALG1] to remove degenerate entanglements. Decomposition of the three-dimensional protein structure into a series of threaded loops closed by native contacts has an inherent degeneracy where two lassos can have significant primary structure overlap and, therefore, should be treated as a single entanglement. Once we have the total set of degenerate loops with crossing events, we run a clustering algorithm to cluster loops with crossing events that are likely part of the same entanglement. The basic idea is to cluster loops with crossing events that are spatially close, and the directionality of piercings is shared. The clustering algorithm is as follows:

1. Identify and sort in ascending order unique crossing residue sets along with their chirality
   - Crossing Residue sets are represented by the vector **r** $(r_1, r_2, \dots, r_n)$
   - *Chirality (direction which the thread twists around the loop)* is the sign of scalar product between the plane vector and piercing vector which can be either negative or positive depending on the orientation of the two vectors. Example: $+r_1, +r_2, \dots, -r_n$
   - *Unique* refers to grouping loops with the same crossing residue vector **r**.

2. Find the minimal loop in each group. In the end, each unique crossing residue set corresponds to a single loop that is the smallest loop size for that set. For example, given n loops that share the crossing vector $(+12, -15, -38, +60)$, the smallest loop size is min$[(j_1 - i_1), (j_2 - i_2), \dots, (j_n - i_n)]$.

3. Merge pairs of minimal loop entanglements with same chirality
   - Case 1: Entanglement pairs with different number of crossing residues
     a. Find entanglements pairs using combinations from step 2 such that an entanglement is not repeated twice. For example, elements ABCD will have the following combinations: AB, AC, AD, BC, BD, and CD.

     b. Loop through the entanglement pairs and merge a pair if the entanglements meet the following criteria:

        - Perform the cartesian product between the crossing residues of one entanglement and those of the other entanglement only if they have the same chirality. Calculate the difference between the products. If any of their distance is less than or equal to 3 residues move to the next criteria. For example, given an entanglement pair A and B with crossing vectors:

          $$r_A = [+r_{A1}, -r_{A2}]$$

          $$r_B = [+r_{B1}, +r_{B2}, +r_{B3}]$$

          Then the cartesian product P is:

          $$P = P(r_A, r_B) = r_A \times r_B = \{(a, b) \mid a \in r_A \; and \; b \in r_B\} = \{p_1, \dots, p_k\}$$

          Where $p_k = (r_{A,h}, r_{B,f})$ and $h \in [1, |r_A|]$ and $f \in [1, |r_B|]$ are dummy variables indexing the crossing residues with the vectors r_A and r_B respectively. Remove elements of P if the signs of the crossing residues don't match.

$$P = \{p_k(a,b) \; \in P \; if \;\; sgn(a) = sgn(b)\}$$

- For each element of P calculate the absolute difference between the crossing residues and if any are less than or equal to 3 then move to the next criteria

c. If the loop of one entanglement (defined by the native contacts $i_A\,, j_A$) overlaps with the loop of the other entanglement (defined by $i_B\,, j_B$) defined as:

$$(i_A \in [i_B, j_B] \; \lor j_A \in [i_B, j_B]) \; \lor \; (i_B \in [i_A, j_A] \; \lor j_B \in [i_A, j_A])$$

Then move to the next criteria.

d. If crossing residues are not in the loop range of both entanglements defined as:

$$r_A \cup r_B \notin [\min(i_A\,, j_A, i_B\,, j_B), \max(i_A\,, j_A, i_B\,, j_B)]$$

Then move to the next criteria.

e. Finally, if the minimum distance of the double cartesian product between entanglements A and $B's$ crossing vectors is less than 20 residues then we remove the entanglement with the least number of crossings. For example:

- Given two crossing vectors:

$$r_A = [+r_{A1}, -r_{A2}] = \; [225, -272]$$

$$r_B = \; [+r_{B1}, +r_{B2}, +r_{B3}] = \; [214, 224, 270]$$

i. Find cartesian product P between crossing residue vectors r_A and r_B:

$$P = P(r_A, r_B) = r_A \times r_B = \{(a,b) \mid a \in r_A \; and \; b \; \in r_B\} = \{p_1, \ldots, p_k\}$$

$$P = P(r_A, r_B) = \begin{bmatrix} (225, 214) & (-272, 214) \\ (225, 224) & (-272, 224) \\ (225, 270) & (-272, 270) \end{bmatrix}$$

ii. Find the double cartesian product D:

$$D = P \times P = \{(p_n, p_m) \mid p_n \in P \; and \; p_m \in P\} = \{d_1, \ldots, d_l\}$$

$$D \; = \begin{bmatrix} [(225, 214), (-272, 214)] & \cdots & [(225, 270), (-272, 214)] \\ \vdots & \ddots & \vdots \\ [(225, 214), (-272, 270)\,] & \cdots & [(225, 270), (-272, 270)] \end{bmatrix}$$

Where $n, m \in [1, k]$ are dummy variables indexing an element of the original product $P = P(r_A, r_B)$ of length k. D then can have elements d_l defined more precisely as:

$$d_l = (p_n, p_m) = \left((r_{A,h}, r_{B,f}), (r_{A,q}, r_{B,s})\right)$$

Where $h, q \in [1, |r_A|]$ and $f, s \in [1, |r_B|]$ are dummy variables indexing the crossing residues with the vectors r_A and r_B respectively.

iii. Remove elements of D that will lead to a trivial zero distance result.

$$D = \{d_l \ \in D \ if \ \ (h \neq q) \vee \ (f \neq s)\}$$

iv. Finally, calculate the Euclidean distance for each element of D and then find the minimum distance. If the minimum distance is less than 20 residues then remove the entanglement with the least number of crossings.

v. *Note:* The algorithm continues even if the second distributive product only has one group. If so, then skip “remove pairs if there is common residue column-wise in that pair” step and moving onto calculating the Euclidean distance for that group. For example, when one entanglement has two crossing residues and the other has one crossing residue assuming these entanglements meet all previous criteria.

▪ Case 2: Entanglement pairs with same number of crossing residues

a. Find entanglements pairs using combinations from step 3 such that an entanglement is not repeated twice. For example, elements ABCD will have the following combinations: AB, AC, AD, BC, BD, and CD.

b. Loop through the entanglement pairs and merge a pair if the entanglements meet the following criteria:

- If the loop of one entanglement (defined by the native contacts i_A , j_A) overlaps with the loop of the other entanglement (defined by i_B , j_B) defined as:

$$(i_A \in [i_B, j_B] \ \vee j_A \in [i_B, j_B]) \ \vee \ (i_B \in [i_A, j_A] \ \vee j_B \in [i_A, j_A])$$

Then move to the next criteria.

- All pairwise distances between crossing residues are less than or equal to 20 residues and they have the same chiralities For example,

  i. Given two crossing vectors:

$$r_A = \ [+r_{A1}, -r_{A2}]$$
$$r_B = \ [+r_{B1}, -r_{B2} \ ]$$

  ii. The distances are: abs($r_{A1} - \ r_{B1}$), abs($r_{A2} - \ r_{B2}$)

  iii. If all distances are less than 20 residues then move to the next criteria

- Remove the entanglement with the larger loop. Avoid an entanglement pair if one of its entanglements or both entanglements has already been merged.

c. *Note*: Order of the combination pair does not matter because there is an implicit order when we merge. I am assuming for the sake of these examples that the entanglements meet the criteria for merging.

- Example: ent1, ent2, ent3. Let's say ent1 loop is larger than ent2 loop and ent2 loop is larger than ent3 loop (ent1 > ent2 > ent3)
- Combination steps:
  1. (ent1, ent2)
  2. (ent1, ent3)
  3. (ent2, ent3)
- The first pair is merged into ent2 and ent1 is eliminated then pair 2 is ignored. We move on to the third entanglement pair (ent2, ent3) and this get merged to ent3 (ent2 is eliminated). The final answer is ent3.

- Another example but same entanglements (ent1 > ent2 > ent3):

  i. If the order of the entanglements were: ent2, ent3, ent1 then combination step will yield:
     - 1. (ent2, ent3)
     - 2. (ent2, ent1)
     - 3. (ent3, ent1)

  ii. In the first pair, ent2 is still eliminated since ent2 loop is larger than ent3 loop. The second pair is ignored. Moving onto the third pair, ent1 is eliminated since ent1 > ent3 and the answer is still ent3.

4. Utilizes a density-based clustering to cluster entanglements with the same number of crossing residues and chiralities. For the entanglement A and B with the same number of crossing residues, the distance between entanglement A (i_A, j_A, r_A) and entanglement B (i_B, j_B, r_B) is computed as follows:

$$d_{(i,j,\boldsymbol{r})^A,\,(i,j,\boldsymbol{r})^B} = \sqrt{(i^A - i^B)^2 + (j^A - j^B)^2 + \sum_{k=1}^{n} (r_k^A - r_k^B)^2}$$

Where n is the length of the vector r_A, r_B.

In this method, the first entanglement or point picked has the largest number of neighbors (i.e., other entanglements) determined using the distance formula where each distance is less than or equal to an optimized threshold (OT). This entanglement and its neighbor are assigned to the same cluster. The process repeats for the entanglement with the next largest number of neighbors and ends until the list of entanglements is exhausted for a gene[1].

As a side note, our OT is obtained by picking an equal number of PDB structures with different structural classes [α, β, α/β] and different protein sizes for *Escherichia coli* (sample size was 100). In *Saccharomyces cerevisiae* and *Homo sapiens*, we sampled random 200 entangled proteins in each proteome. These proteins are then evaluated using the silhouette score at different thresholds from 1 to 701. Afterward, we average all silhouette scores for a given threshold across our sample proteins. The maximum average score across our thresholds and sample proteins is chosen as the OT.

The silhouette score measures the similarity between points in a cluster compared to other clusters, where $\mathrm{a_p}$ is the average intra-cluster distance and $\mathrm{b_p}$ is the average inter-cluster distance of point p.

$$s_p = \frac{b_p - a_p}{max\ (a_p, b_p)}$$

The average silhouette score is the arithmetic mean of the silhouette score from each structure at a given threshold. OT for *Escherichia coli*, *Saccharomyces cerevisiae*, and *Homo sapiens* were 57, 49, and 52, respectively. Plotting the average silhouette scores reveals that the maximum is where the graph peaks as shown in Figure S1.

Finally, a representative entanglement from each cluster is obtained based on the geometric median of the crossings and loop size. Specifically, the geometric median of the crossings is an optimization problem where the goal is to minimize the sum of distances for crossing residues. In a cluster, the geometric median for crossings $(r_1, \ldots, r_n)$ was calculated as

$$\boldsymbol{r}_{GM} = argmin \sum_{y=1}^{n} \left\| \left(min \sum_{i=1}^{n} \|r_y - x_i\| \right) - r_y \right\|$$

where x_i represents individual points and *n* represents the number of crossing residues. The entanglement whose crossings are spatially close to this geometric median and has the smallest loop is chosen as the representative entanglement for a given cluster. This process repeats for the remaining clusters. For most proteins, density-based clustering is often unnecessary since step 3 already outputs unique entanglements. We keep the density-based clustering for those proteins that do not output unique entanglements after step 3.

To check the performance of the clustering algorithm, we compare the unique crossing residue vectors per gene between the raw entanglements and the clustered entanglements as shown in Figure S2. The figure shows that both distributions are nearly equal, indicating that we are not over-clustering or under-clustering.

Sampling Algorithm [ALG2] to permute crossing residues for each entanglement. Enumerating the set of valid permutations P is difficult due to the computational intractability of modeling all possible 3-dimensional structures of permuted sequences subject to given $C\alpha$ coordinates. One necessary condition for a crossing residue is that it has to be located in a buried part of the protein, due to the entanglement's topological property. Another important condition was to sample crossing residues based on the spatial orientation from the observed data. These conditions are incorporated in the algorithm detailed below, which is used to sample valid crossing residue positions given the protein's topology.

1.Obtain the crossings for the entanglement

2. Obtain a random distance matrix from the population of all distance matrices derived from entanglements with the same number of crossings across the proteome.

    a. The population of distance matrices was obtained by performing the Euclidean pairwise distances between the $C\alpha$ coordinates of the crossing residues within an entanglement. This is performed for every entanglement in a gene and for every gene in the proteome. *Note* that Euclidean pairwise distance is not performed if an entanglement has a single crossing.

3. Sampling the placement of the first new crossing residue:

    a. Find the set difference between all of the protein residues and those that were selected before as the first sampled residue. The algorithm can reinitialize to the beginning of step 3, so it is important to select protein residues without replacement.

        - *Note:* The sampling algorithm will reinitialize to beginning of step 3 if cannot find successful placements for all the crossings in an entanglement, but the first sampled residue is not selected again unless successful placements were not possible because all protein residues and distances in the matrix have been exhausted.

    b. Perform a stochastic algorithm using the previous step as input.

        - The stochastic algorithm performs if a random float from 0 to 1 inclusive is less than or equal to the solvent exposure probability for a random residue from the input then that residue is our first sampled residue for the entanglement. The algorithm continues if it cannot find the first sampled residue until all input residues are exhausted. If all input residues are exhausted and the algorithm cannot find the first sampled residue then reinitiate step 3 from the beginning. Please see the *Note* for more details. An equation for solvent exposure probability is obtained from several steps:

            i. For each gene individually in the proteome the probability distribution for the ratio of the surface accessible solvent area (SASA) of the crossings to the average SASA for the protein was generated with a bin size of 0.01.

            ii. Second, an equation was obtained by fitting the histogram to an exponential function (see Figure S3).

            iii. Lastly, the ratio of the SASA of the random residue to the average SASA for the protein was calculated for every residue in the protein. This ratio was used with the equation to sample new crossing residues that match the shape of the distribution,

        - *Note*: In case the sampling algorithm reinitializes to the beginning of step 3 then pick a new random distance matrix from the population if all protein residues were sampled and all distances in the matrix were used. Keep track of the number of distance matrices being randomly picked because if this number equals the number of distance matrices available for the number of crossings in step 1 then skip the gene and move onto the next gene. The reason you skip the gene is because all the distances matrices available for the number of crossings have been exhausted and all protein residues were selected.

4. For sampling the n^{th} new crossing residue where $n > 1$:

a. Randomly pick (without replacement) $n - 1$ distance(s) from the random distance matrix selected in step 2:

   - For example, there is an entanglement with three crossings so the number of distances in the distance matrix is 3. When $n = 2$, then a single distance is picked from the matrix. In later steps, we remove that distance if a criterion is met. When $n = 3$, the remaining two distances are picked from the matrix.

b. Keep track of distances randomly picked from the matrix in case placement of nth crossing residue fails.

c. Create thresholds of plus-and-minus 4 Angstrom for each distance randomly selected from the matrix.

d. Create spherical shells using thresholds.

   - For each threshold:

      i. Use its index in the list of thresholds to grab the previous sampled residue and its Cα coordinate.

      ii. Calculate the Euclidean distance between the previous sampled residue and all protein residues.

      iii. Find the distances that fall within the threshold (ignore distance = 0) and correspond them with protein residues. If successful, then keep track of those protein residues for the next step and move to the next threshold.

      iv. Otherwise, sampling algorithm reinitializes to the beginning of step 3 with the same distance matrix originally picked and removes previous sampled crossing residues if any.

e. If there are protein residues within our thresholds from previous step, then start sampling:

   - Find the set intersection between the residues in each sphere

   - Find the set difference between the intersection from previous step and crossing residues that were sampled already

   - If both previous steps are successful then input the difference from previous step into the stochastic algorithm to pick a residue. If successful, then keep track of the sampled residue and delete the distance from the distance matrix that corresponds to the residue. If not successful, then repeat the step.

   - However, if the first two steps are not successful (i.e., there is no set intersection or there is no set difference) then the sampling algorithm reinitializes to the beginning of step 3 with the same distance matrix originally picked and removes previous sampled crossing residues if any.

5. Move to the next entanglement for the gene.

6. In the end, the output consists of sampled crossing residues for each entanglement for the gene.

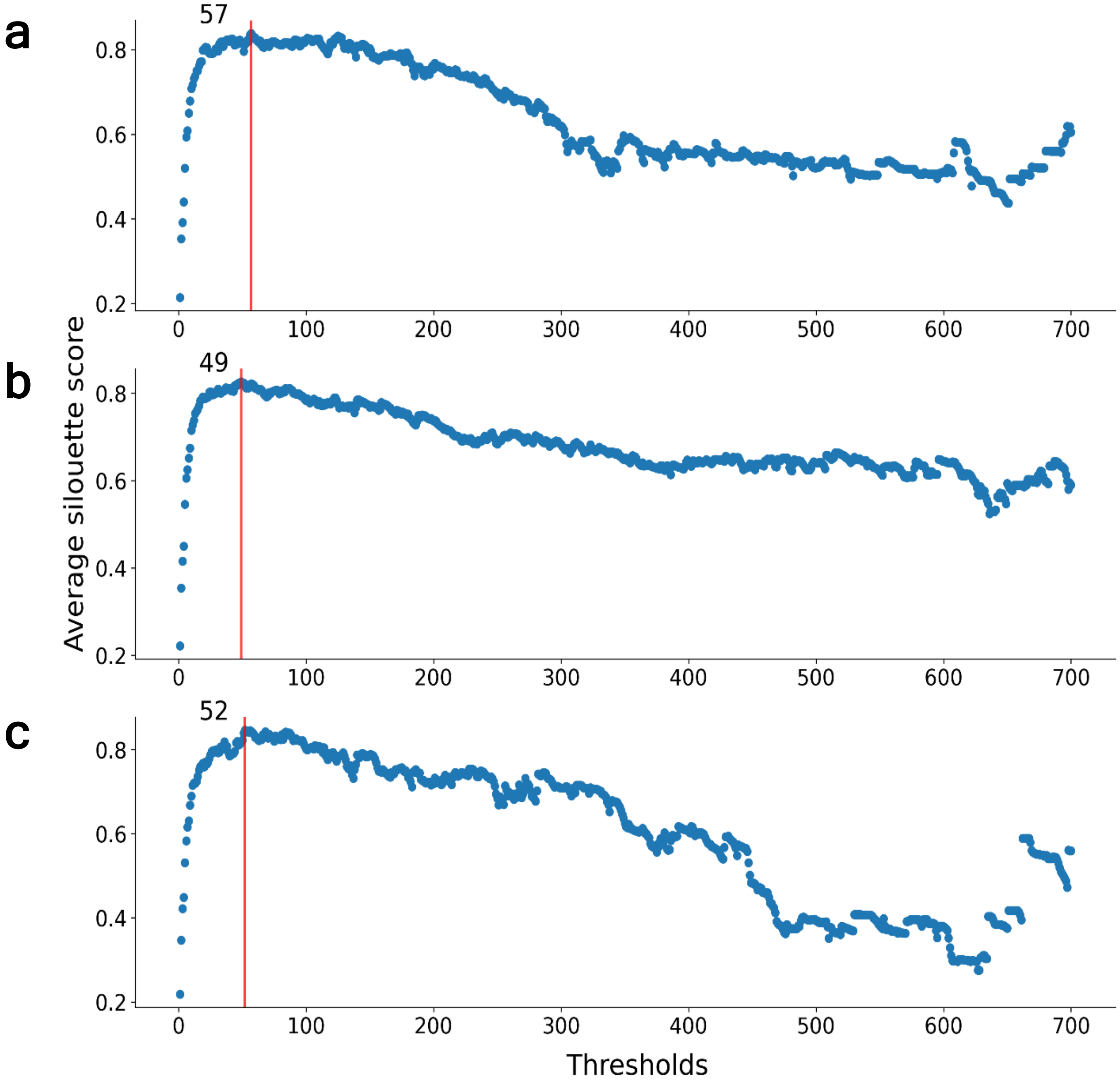

Figure S1. Optimized thresholds obtained using average silhouette score. a-c Scatter plot of (**a**) *Escherichia coli*, (**b**) *Saccharomyces cerevisiae*, and (**c**) *Homo sapiens* representing the average silhouette score from thresholds 1 to 701. The red line in each plot is the maximum average silhouette score or the OT.

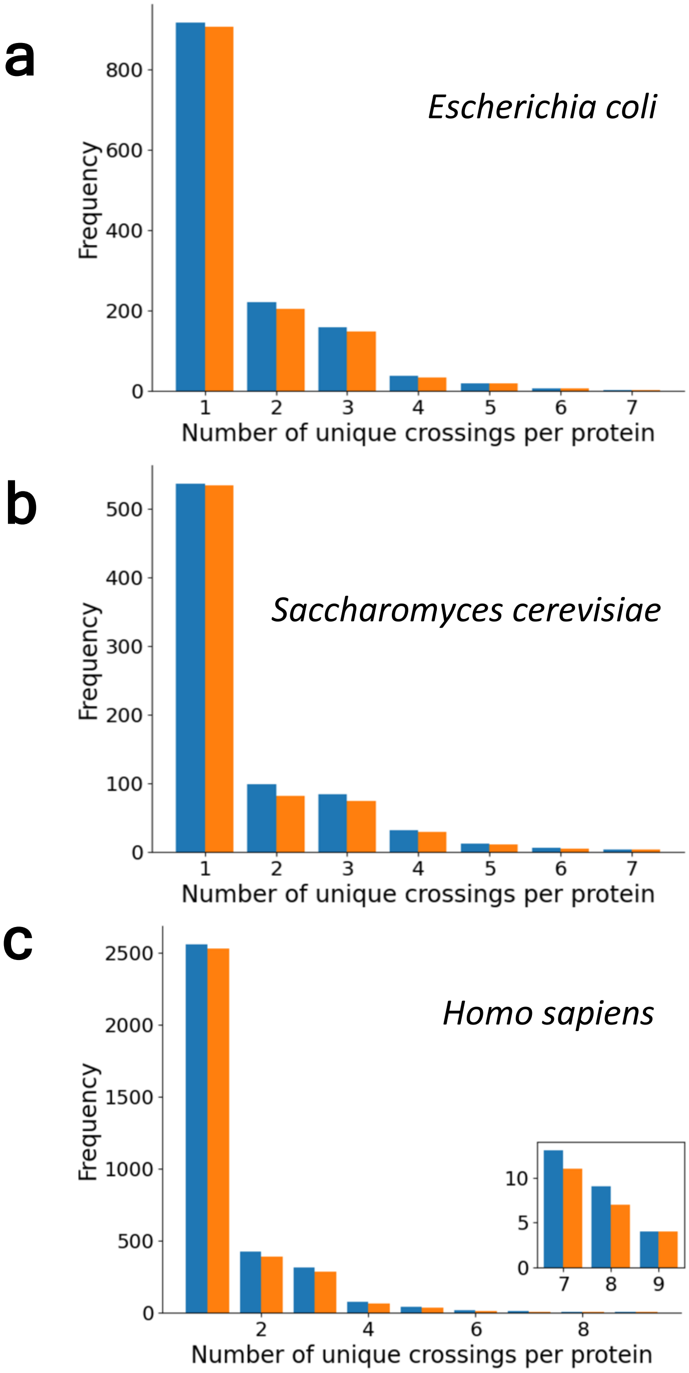

Figure S2. Evaluation of clustering algorithm performance. a-c Frequency plots of (**a**) *Escherichia coli*, (**b**) *Saccharomyces cerevisiae*, and (**c**) *Homo sapiens* comparing the raw entanglements (in blue) to the clustered entanglements (in orange). The x-axis represents the number of entanglements with unique crossings per protein. These results demonstrate that the clustering does not change the distribution of entanglement properties.

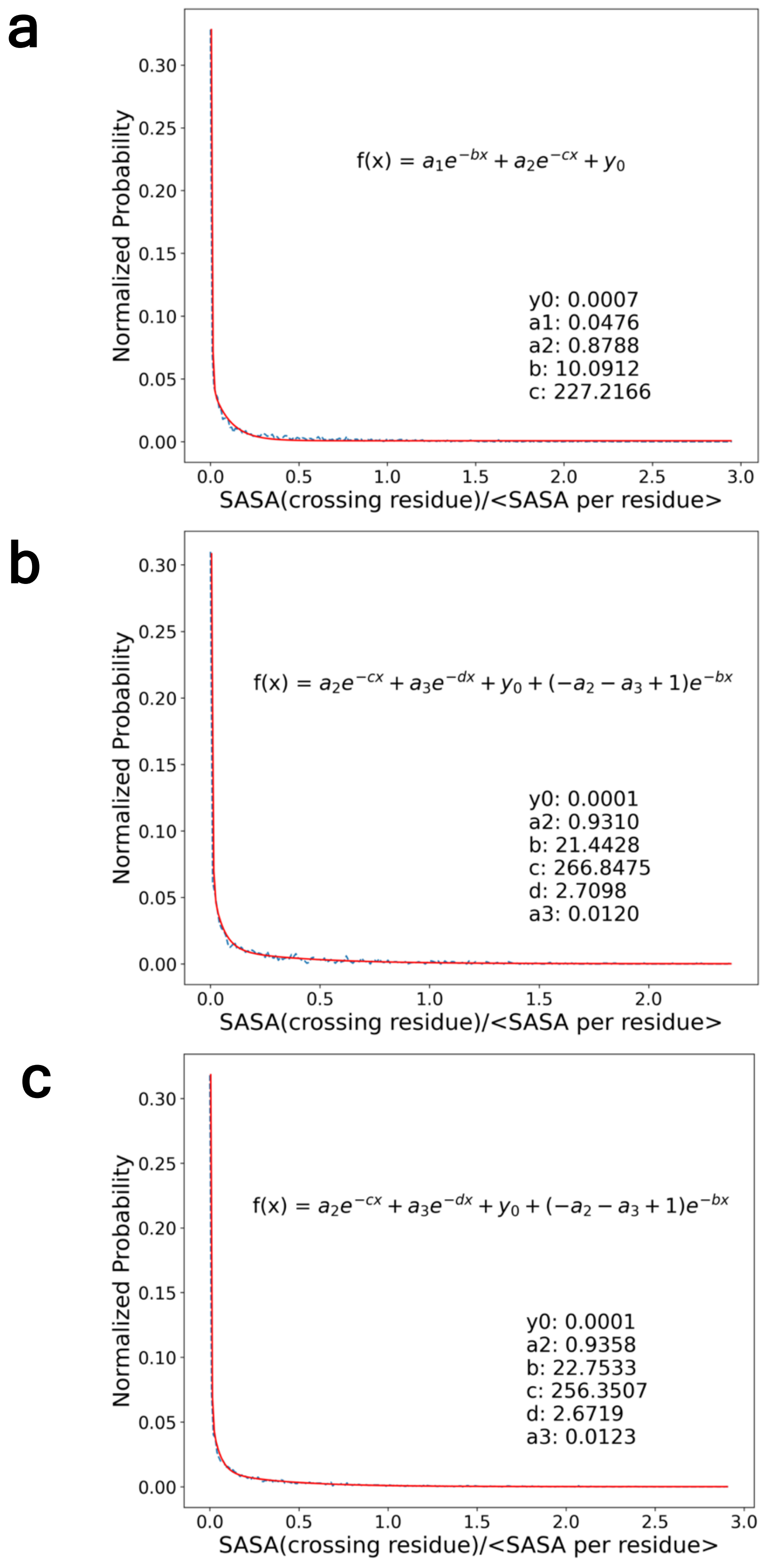

Figure S3. Probability Distribution of Buried Crossing Residues. a-c Exponential plots of (**a**) *Escherichia coli*, (**b**) *Saccharomyces cerevisiae*, and (**c**) *Homo sapiens* representing the ratio of the surface accessible solvent area (SASA) of the crossings to the average SASA for the protein shown as blue dashes. The red curve is fitted to the histogram with a bin width of 0.01.

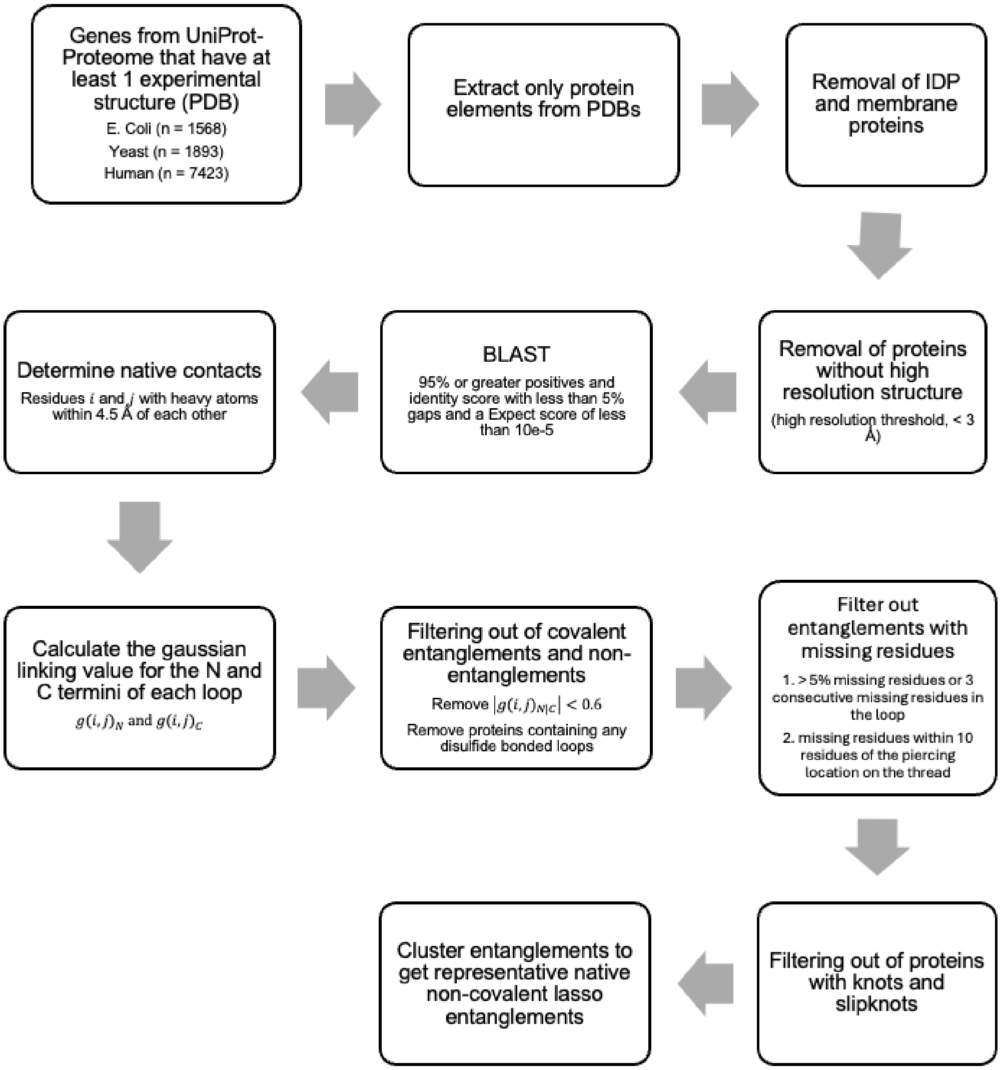
Genes from UniProt-Proteome that have at least 1 experimental structure (PDB)
E. Coli (n = 1568)
Yeast (n = 1893)
Human (n = 7423)
Extract only protein elements from PDBs
Removal of IDP and membrane proteins
Removal of proteins without high resolution structure
(high resolution threshold, < 3 Å)
BLAST
95% or greater positives and identity score with less than 5% gaps and a Expect score of less than 10e-5
Determine native contacts
Residues i and j with heavy atoms within 4.5 Å of each other
Calculate the gaussian linking value for the N and C termini of each loop
$g(i,j)_N$ and $g(i,j)_C$
Filtering out of covalent entanglements and non-entanglements
Remove $|g(i,j)_{N|C}| < 0.6$
Remove proteins containing any disulfide bonded loops
Filter out entanglements with missing residues
1. > 5% missing residues or 3 consecutive missing residues in the loop
2. missing residues within 10 residues of the piercing location on the thread
Filtering out of proteins with knots and slipknots
Cluster entanglements to get representative native non-covalent lasso entanglements

Figure S4. Main Steps in Entanglement Identification. Flowchart describes an overview of our methods including detection of entanglements. Please refer to the Methods for exact details and/or visit our GitHub.

Spatial Association Frequency Table for *Escherichia coli* Non-Covalent Lasso Entanglements (uncorrected p-values)

Functions	Enrichment	Depletion	Neither	Total
DNA binding	5	0	25	30
RNA binding	2	1	28	31
Zinc finger region	0	0	4	4
Active site	21	8	257	286
Protein - protein interfaces	27	15	424	466
Metal binding	11	5	145	161
Small molecules	66	16	494	576
All	93	22	690	805

Spatial Association Frequency Table for *Saccharomyces cerevisiae* Non-Covalent Lasso Entanglements (uncorrected p-values)

Functions	Enrichment	Depletion	Neither	Total
DNA binding	0	0	17	17
RNA binding	4	1	28	33
Zinc finger region	0	1	16	17
Active site	12	7	103	122
Protein - protein interfaces	22	17	224	263
Metal binding	2	12	73	87
Small molecules	28	19	224	271
All	48	25	364	437

Spatial Association Frequency Table for *Homo sapiens* Non-Covalent Lasso Entanglements (uncorrected p-values)

Functions	Enrichment	Depletion	Neither	Total
DNA binding	7	2	58	67
RNA binding	14	2	45	61
Zinc finger region	0	6	42	48
Active site	34	18	722	774
Protein - protein interfaces	87	41	922	1050
Metal binding	44	16	369	429
Small molecules	199	60	1340	1599
All	263	80	1828	2171

Table S1. **Raw Results of Structural Enrichment Analysis. a-c** Counts of (**a**) *Escherichia coli*, (**b**) *Saccharomyces cerevisiae*, and (**c**) *Homo sapiens* genes that are enriched, depleted and neither are listed in the table without FDR hypothesis correction. The last row addresses the question, "are non-covalent lasso entanglements near any functional residues"? The genes representing the counts can be found in SI File 8.