

**Computational Tools for Protein-Protein Interaction Prediction and Analysis**

by

Eric William Bell

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
(Bioinformatics)  
in the University of Michigan  
2022

Doctoral Committee:

Associate Professor Peter L. Freddolino, Chair  
Professor Philip C. Andrews  
Research Assistant Professor Matthew O'Meara  
Professor Gilbert S. Omenn  
Associate Professor Evan S. Snitkin

Eric W. Bell

[ewbell@umich.edu](mailto:ewbell@umich.edu)

ORCID iD: 0000-0002-3419-4398

© Eric W. Bell 2022

## **Dedication**

This dissertation is dedicated to my family and friends, whose support and love has made me into the scientist I am today.

## **Acknowledgements**

There are a great deal of people who I would like to thank for their support and guidance; without them I would not have achieved. First, I would like to thank Dr. Lisa Ryno for inspiring my love of research and allowing me the freedom to pursue my computational research interests. To my current and former members of my committee, thank you for your guidance and support in developing my project into a mature body of scientific work.

I would also like to thank Chengxin Zhang and Wallace Chan for helping me gain my footing in the field of structural bioinformatics. I would not have been as successful as I have been without your help in learning the tools and logistics of working in a fully computational lab. Also, I would like to thank my friends in the Freddolino lab: Rucheng, Catherine, Lara, Manasa, Anna, Xi, Jeremy, Chase, Willow, Ali, Yulduz, Rebecca thank you all. I would particularly like to thank Jake Schwartz for being a great “partner in crime” and a fun companion during my time in the lab.

Peter, I owe so much of my success to your willingness to support me after the dissolution of my former lab. I know that the amount of extra work you have shouldered to keep things running smoothly for all of us has been enormous, yet your dedication to doing the right thing has been incredibly helpful and admirable. I have always left our conversations feeling like your insight was compassionate and helpful. I sincerely hold you as a role model for what a professor should be.

Thank you to my friends who have supported me throughout my time in school; the memories I have made with you all I will carry with me for the rest of my life. In particular I’d

like to thank Ajay Larkin, whose close friendship has been a major source of motivation for me and kept my work-life balance in check.

Finally, I would like to thank my family. Despite my research being out of their comfort zone, my parents have been supportive and loving through my entire career and have given invaluable advice on how to approach the difficulties I faced during graduate school. I would also like to thank my sister Mandy, whose advice and encouragement based on her own graduate school experience was essential to my success.

## Table of Contents

|                                                                                                                                                                      |      |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| Dedication .....                                                                                                                                                     | ii   |
| Acknowledgements.....                                                                                                                                                | iii  |
| List of Tables .....                                                                                                                                                 | ix   |
| List of Figures .....                                                                                                                                                | x    |
| Abstract .....                                                                                                                                                       | xiii |
| Chapter 1 Introduction .....                                                                                                                                         | 1    |
| 1.1 Importance of protein function.....                                                                                                                              | 1    |
| 1.2 Protein structure and function prediction .....                                                                                                                  | 3    |
| 1.2.1 Protein structure prediction .....                                                                                                                             | 3    |
| 1.2.2 Protein function annotation and prediction .....                                                                                                               | 5    |
| 1.3 Protein-protein interactions .....                                                                                                                               | 6    |
| 1.3.1 Experimental methods for PPI determination .....                                                                                                               | 7    |
| 1.3.2 PPI Prediction .....                                                                                                                                           | 8    |
| 1.4 Other Published Works .....                                                                                                                                      | 11   |
| 1.5 Summary .....                                                                                                                                                    | 11   |
| 1.6 References .....                                                                                                                                                 | 12   |
| Chapter 2 PEPPI: Whole-proteome Protein-protein Interaction Prediction through Structure and Sequence Similarity, Functional Association, and Machine Learning ..... | 19   |
| 2.1 Abstract .....                                                                                                                                                   | 19   |
| 2.2 Introduction .....                                                                                                                                               | 20   |
| 2.3 Results .....                                                                                                                                                    | 22   |

|                                                                                               |    |
|-----------------------------------------------------------------------------------------------|----|
| 2.3.1 Pipeline overview and module cross-validation .....                                     | 22 |
| 2.3.2 PEPPi benchmark and performance .....                                                   | 23 |
| 2.3.3 SARS-CoV-2 and human interactome modeling .....                                         | 26 |
| 2.4 Discussion & conclusion .....                                                             | 29 |
| 2.5 Methods .....                                                                             | 30 |
| 2.5.1 PPI prediction by SPRING threading (“SPRING”/”SPRINGNEG” module) .....                  | 30 |
| 2.5.2 PPI prediction by sequence interology (“SEQ” module) .....                              | 33 |
| 2.5.3 PPI Prediction by non-similarity features (“CT”/”STRING”) .....                         | 34 |
| 2.5.4 Pipeline training and benchmarking .....                                                | 36 |
| 2.5.5 SARS-CoV-2 virus and human host protein sequence collection .....                       | 37 |
| 2.6 Acknowledgements .....                                                                    | 38 |
| 2.7 Figures .....                                                                             | 38 |
| 2.8 Tables .....                                                                              | 41 |
| 2.9 References .....                                                                          | 42 |
| Chapter 3 Probiotic-Host Interaction Network Modeling and Virtual Reality Visualization ..... | 46 |
| 3.1 Abstract .....                                                                            | 46 |
| 3.2 Introduction .....                                                                        | 46 |
| 3.3 Results .....                                                                             | 48 |
| 3.3.1 E. coli Nissle interaction network .....                                                | 48 |
| 3.3.2 L. rhamnosus interaction network .....                                                  | 52 |
| 3.3.3 FALCON Visualization of probiotic networks .....                                        | 55 |
| 3.4 Discussion and Conclusion .....                                                           | 57 |
| 3.5 Methods .....                                                                             | 58 |
| 3.5.1 Sequence collection and PEPPi classification .....                                      | 58 |
| 3.5.2 GO Term analysis .....                                                                  | 60 |

|                                                                                                                               |    |
|-------------------------------------------------------------------------------------------------------------------------------|----|
| 3.5.3 FALCON Implementation and Features .....                                                                                | 61 |
| 3.6 Acknowledgements .....                                                                                                    | 62 |
| 3.7 Figures .....                                                                                                             | 63 |
| 3.8 References .....                                                                                                          | 67 |
| Chapter 4 Evaluation of Multiple Sequence Alignment Pairing Schemes for Interchain<br>Coevolutionary Structure Modeling ..... | 70 |
| 4.1 Abstract .....                                                                                                            | 70 |
| 4.2 Introduction .....                                                                                                        | 70 |
| 4.3 Results .....                                                                                                             | 72 |
| 4.3.1 Comparison of interchain distance predictors .....                                                                      | 72 |
| 4.3.2 Paired MSA generation benchmarking .....                                                                                | 74 |
| 4.4 Discussion and Conclusion .....                                                                                           | 75 |
| 4.5 Methods .....                                                                                                             | 76 |
| 4.5.1 Dataset collection and benchmark .....                                                                                  | 76 |
| 4.5.2 MSA pairing methods.....                                                                                                | 77 |
| 4.6 Acknowledgements .....                                                                                                    | 79 |
| 4.7 Figures .....                                                                                                             | 80 |
| 4.8 Tables .....                                                                                                              | 82 |
| 4.9 References .....                                                                                                          | 83 |
| Chapter 5 Conclusion and Future Directions.....                                                                               | 85 |
| 5.1 Experimental validation of computational approaches .....                                                                 | 85 |
| 5.2 Protein dimer structure modeling .....                                                                                    | 86 |
| 5.3 Multimeric and multi-omic interaction prediction .....                                                                    | 88 |
| 5.4 Final Conclusion.....                                                                                                     | 89 |
| 5.5 References .....                                                                                                          | 90 |

|                                                                                                                                        |     |
|----------------------------------------------------------------------------------------------------------------------------------------|-----|
| Appendix A: DockRMSD: an open-source tool for atom mapping and RMSD calculation of symmetric molecules through graph isomorphism ..... | 92  |
| A.1 Abstract.....                                                                                                                      | 92  |
| A.2 Introduction .....                                                                                                                 | 93  |
| A.3 Implementation.....                                                                                                                | 96  |
| A.4 Results and discussion.....                                                                                                        | 98  |
| A.4.1 Docking conformation dataset and generation .....                                                                                | 98  |
| A.4.2 Docking RMSD calculation through DockRMSD .....                                                                                  | 99  |
| A.4.3 DockRMSD runtime comparison .....                                                                                                | 102 |
| A.5 Conclusion.....                                                                                                                    | 104 |
| A.6 Acknowledgements .....                                                                                                             | 105 |
| A.7 Figures .....                                                                                                                      | 106 |
| A.8 Tables .....                                                                                                                       | 110 |
| A.9 References .....                                                                                                                   | 111 |

## List of Tables

|                                                                                                                                                                                                                                                                                                                                                                             |     |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| <b>Table 2.1</b> Performance of the control pipelines against PEPPI in the balanced benchmark. P-values (calculated by 1000-iteration bootstrapping of the distribution of differences in performance between methods) are presented in parentheses. ....                                                                                                                   | 41  |
| <b>Table 2.2</b> Performance of the control pipelines against PEPPI in the unbalanced benchmark. P-values (calculated by 1000-iteration bootstrapping of the performance difference distribution) are presented in parentheses. ....                                                                                                                                        | 42  |
| <b>Table 4.1</b> Average MAE of each benchmark program for the top-L/5, L/2, and L distances predicted (where L is the sum of the two chain lengths). Ground truth-based MAEs are labeled as “GT”. The total number of completed predictions considered in the average is indicated in parentheses (GT contains one additional point).....                                  | 82  |
| <b>Table 4.2</b> Average MAE of each MSA pairing strategy for the top-L/5, L/2, and L distances predicted by AlphaFold2 (where L is the sum of the two chain lengths). Ground truth-based MAEs are labeled as “GT”. The total number of completed predictions considered in the average is indicated in parentheses (GT contains one additional point for SPEC and CT)..... | 83  |
| <b>Table A.1</b> Counts of 3,430 total RMSD calculations whose error relative to the deterministic DockRMSD calculation is zero, small (nonzero but smaller than 2.0Å), or large (greater than 2.0Å). ....                                                                                                                                                                  | 110 |
| <b>Table A.2</b> A contingency table for 343 RMSD calculations between docked ligand poses and their respective native crystal structure ligand poses, calculated both by DockRMSD and the Hungarian algorithm. ....                                                                                                                                                        | 111 |

## List of Figures

**Figure 2.1** An illustration of the PEPPI pipeline. This pipeline functions by analyzing a pair of input sequences via a series of independent modules, including structure similarity, sequence similarity, neural network classification, and functional association data. These modules are combined using a naïve Bayesian consensus classifier, which provides the final interaction score as a log-likelihood ratio. .... 38

**Figure 2.2** PEPPI benchmark results. (a) 10-fold cross validation AUROC reveals that the full PEPPI pipeline outperforms its component modules: the neural network classifier (CT), the functional association data (STRING), the sequence similarity method (SEQ), the structure similarity method (SPRING), and the non-interaction similarity method (SPRINGNEG). (b) An ROC curve of the performance of PEPPI against PRISM, a structure similarity-based method, SPRINT, a sequence similarity-based method, PIPR, a deep learning-based method, and D-SCRIPT, a structure-aware deep learning-based method on a balanced testing set. The dotted line represents the performance of random classification. (c) Precision-recall curve of the performance of PEPPI against several other comparable programs on an unbalanced testing set. The dotted line represents the performance of random classification. (d) A superposition of an example dimer model (PDB 3CI0; chain J in red, chain K in blue) on its dimer template structure (PDB 5VTM; chain W in yellow, chain X in cyan). .... 39

**Figure 2.3** A summary of SARS-CoV-2/human interactome prediction. (a) A network overview of the full interactome of human/SARS-CoV-2 protein pairs, with SARS-CoV-2 proteins colored red and human proteins colored blue. (b) A bar chart of the number of predicted interactions involving each SARS-CoV-2 protein. Proteins which were not predicted to have any interactions were excluded. (c) A superposition of a dimer model of the top-ranked SARS-CoV-2 Spike (in red) and Human ACE2 (in blue) interaction on its dimer template structure (PDB 6ACG; chain D in cyan, chain C in yellow). (d) A superposition of a dimer model of the SARS-CoV-2 nsp3 macrodomain (in red) and human PARP15 macrodomain (in blue) on its dimer template structure (PDB 2W2G; chain A in cyan, chain B in yellow). (e) A superposition of a dimer model of a domain of the SARS-CoV-2 nsp3 (in red) and human NEDD8 (in blue) on its dimer template structure (PDB 5WFI; chain C in cyan, chain A in yellow). (f) A superposition of a dimer model of a domain of the SARS-CoV-2 nsp3 (in red) and human UBD (in blue) on its dimer template structure (PDB 6BI8; chain C in cyan, chain A in yellow). .... 40

**Figure 2.4** An illustration of the domain division procedure for structure-based interology. In this figure, query sequence A ( $Q_A$ ) is divided into domains according to the domain boundary derived from the alignments of its threading templates. After domain division, all domains of sequence A are paired with the single-domain query sequence B ( $Q_B$ ) and analyzed individually. .... 41

**Figure 3.1** Summary of the EcN interaction network. All blue color corresponds to human proteins, and all red color corresponds to bacterial proteins. Note that the shown structures are derived through threading and therefore structure quality is not necessarily indicative of interaction confidence. (a) The EcN network structure. Inset in this structure is the bacterial node degree distribution for the top 30 bacterial hubs. (b) PEPPi structure of CyoA-COX1, the top-ranked interaction. (c) PEPPi structure of HtrE-CEP192. (d) PEPPi structure of YgeR-PLAT3. (e) PEPPi structure of AtsA-ARSB. .... 63

**Figure 3.2** Volcano plots of GO term enrichment for the EcN (left) and LGG (right). P-values are derived from a hypergeometric distribution; Q-values are calculated from these P-values by the Benjamini-Hochberg correction. Points are scaled in size corresponding to the number of protein pairs annotated with the given GO term. .... 64

**Figure 3.3** Summary of the LGG interaction network. All blue color corresponds to human proteins, and all red color corresponds to bacterial proteins. Note that the shown structures are derived through threading and therefore structure quality is not necessarily indicative of interaction confidence. (a) The LGG network structure. (b) Bacterial node degree distribution for the top 30 bacterial hubs. (c) PEPPi structure of aminopeptidase-ERAP1, the top-ranked interaction. (d) PEPPi structure of cell surface protein-TNFRSF12A. (e) PEPPi structure of lipoprotein-serrotransferrin. .... 65

**Figure 3.4** FALCON visualization of the EcN interaction network. Blue nodes correspond to human proteins, red nodes correspond to bacterial proteins. (a) A view of the entire network visualized in 3D. (b) Pseudo-cylindrical feature visualized in FALCON side-by-side with its traditional two dimensional visualization. (c) Ecotin and YdgD clusters visualized in FALCON side-by-side with the traditional visualization of these clusters. In the traditional view, Ecotin and YdgD have been highlighted in yellow. .... 66

**Figure 3.5** FALCON visualization of the LGG interaction network. Blue nodes correspond to human proteins, red nodes correspond to bacterial proteins. (a) A view of the entire network visualized in 3D. (b) Pseudo-cylindrical feature visualized in FALCON next to its traditional two dimensional visualization. (c) Overlapping cluster pair feature visualized in FALCON next to its traditional two dimensional visualization. .... 67

**Figure 4.1** Comparisons of the interchain distance prediction mean absolute error (MAE) for each of the tested benchmark programs. Rows correspond to each pairing between the three programs, columns correspond to different amounts of residue pairs considered in each point's MAE (where L is the sum of the two chain's lengths); the ground truth-based MAE is labeled as "GT". Each plot is marked with the total number of points above and below (and on for the GT column) the dotted line and a p-value calculated by a two-sided Wilcoxon signed-rank test. .... 80

**Figure 4.2** Comparisons of the interchain distance prediction mean absolute error (MAE) for each of the tested MSA pairing schemes against the random baseline. Rows correspond to each pairing between the three programs and randomness, columns correspond to different amounts of residue pairs considered in each point's MAE (where L is the sum of the two chain's lengths); the ground truth-based MAE is labeled as "GT". Each plot is marked with the total number of

points above and below (and on for the GT column) the dotted line and a p-value calculated by a two-sided Wilcoxon signed-rank test..... 81

**Figure 4.3** Comparisons of the interchain distance prediction mean absolute error (MAE) for each of the tested MSA pairing schemes. Rows correspond to each pairing between the three programs, columns correspond to different amounts of residue pairs considered in each point's MAE (where L is the sum of the two chain's lengths); the ground truth-based MAE is labeled as "GT". Each plot is marked with the total number of points above and below (and on for the GT column) the dotted line and a p-value calculated by a two-sided Wilcoxon signed-rank test..... 82

**Figure A.1** Examples of (a) an asymmetric ligand (PDB Ligand ID: CFF); (b) a slightly symmetric ligand (PDB Ligand ID: IBP); (c) a highly symmetric ligand (PDB Ligand ID: QN3). (d) An example ligand structure (left) and the resulting ligand structure when the atoms are reordered according to the optimal query-template atomic correspondence generated by the Hungarian method (right). Since the Hungarian method only takes atom type into account and not the bonds between atoms, the hypothetical molecule proposed by the Hungarian correspondence is physically impossible. .... 106

**Figure A.2** The DockRMSD algorithm. DockRMSD calculates the optimal atom mapping and RMSD value for any given pair of poses for the same ligand, input as a pair of MOL2 structure files..... 107

**Figure A.3** (a) Ligand RMSD calculated by DockRMSD versus that by the naïve RMSD calculations on 1321 ligand molecules with symmetric structures. (b) An example pair of poses where naïve RMSD calculation failed to provide the optimal RMSD due to molecular symmetry (Ligand PDB ID: E10; Receptor PDB ID: 1H22 [24]). Interpose correspondence between oxygen atoms is drawn to represent the source of the RMSD disagreement by different methods. .... 108

**Figure A.4** (a) Comparison of the Hungarian algorithm against DockRMSD for the 3112 molecules whose RMSD was underestimated by the Hungarian algorithm. (b) Comparison of the Hungarian algorithm against DockRMSD for the 190 molecules whose RMSD was underestimated by the Hungarian algorithm in the native ligand pose benchmark. (c) An example pair of poses where the Hungarian algorithm grossly overcorrected for symmetry due to its insensitivity to global molecular topology (Ligand PDB ID: BEG; Receptor PDB ID: 1D4I [26]). Interpose correspondence between central carbon atoms and nitrogen atoms is drawn to represent the source of the RMSD disagreement. Hungarian correspondence is drawn in red to demonstrate that the correspondence should not be allowed according to the chemical inequivalence of the atoms bonded to each atom of the pair. .... 109

**Figure A.5** Box and whisker plots of the walltime distributions (in  $\log_{10}(\text{sec})$ ) for each of the 3430 RMSD calculations as calculated by symmetry-corrected DockRMSD, symmetry-uncorrected naïve RMSD, and symmetry-corrected obrms..... 110

## **Abstract**

In order to understand the basis for diseases at a molecular level, and therefore develop therapeutics for those diseases, we must first understand the function of the proteins impacted by those diseases. An important task in determining these protein functions is determining the complex network of protein-protein interactions (PPIs) which occur in many biological systems. While experimental techniques exist to determine PPIs, they are either too costly to determine whole-proteome PPI networks or have high error rates which make the dataset unreliable. Therefore, computational PPI prediction and analysis methods can be utilized to complement and refine these experimental datasets. In this thesis, we present a PPI prediction algorithm, PEPPI, which leverages structural and sequence homology, functional association, and machine learning to perform whole-proteome PPI prediction; we then benchmark this program against several competing methods. PEPPI is used to predict several cross-species interaction networks, including the SARS-CoV-2 virus-host network as well as the microbe-host interaction networks for a pair of probiotic strains of bacteria. These networks are then analyzed for novel biological insights. To facilitate network-level analysis, we visualize the probiotic networks in a three dimensional network visualizer we developed. Finally, to support future interchain coevolution-based structure modelers and PPI classifiers, we explore several methods of generating paired multiple sequence alignments. Through the contributions we make to the field of PPI prediction in this thesis, we take steps towards furthering our ability to understand and treat disease.

## **Chapter 1 Introduction**

A major component to our understanding of the cell is the protein, the endpoint of the central dogma of molecular biology and the primary functional unit of cells. Proteins are responsible for a wide variety of cellular tasks including metabolic catalysis, cellular structure, biomolecule synthesis, transport, degradation, extracellular recognition and signaling, control of chromosomal structure and gene expression, etc. Due to revolutions in the field of genomic sequencing, protein amino acid sequences can be inferred from genetic features at unprecedented rates, yet the functions of an overwhelming majority of these proteins remain unstudied and poorly understood. At the current time of writing, the Uniprot database[1] contains ~230 million protein sequences from ~1.3 million species, yet only ~568,000 (0.25%) of those proteins from ~14,000 species belong to Swiss-Prot, the manually reviewed and annotated portion of the Uniprot database. This gap is largely due to the lack of high-throughput experimental methods to assess protein function. Therefore, if we are to understand the behavior of proteins, and by extension biological phenomena in cells, we must rely on computational methodologies that can predict protein function from sequence.

### **1.1 Importance of protein function**

At its core, disease is a misfunction of the biological systems present in one's cells. Therefore, if one is to diagnose such diseases on a molecular level, one must first have a clear perspective of a cell's normal functional state. Much effort has been placed upon studying the functional state of cells through the lens of gene expression, as many cell states (including disease states) can be readily described through identification of which genes are being

expressed and the magnitude to which they are expressed. In the case of inherited genetic disease, many of these gene expression profiles have been traced back to genetic variants which are associated with the resulting expression state,[2] thus allowing us to predict the occurrence of disease using only genomic information. However, not all disease is due to variants, as some disease states can be caused by chronic exposure to unfavorable environments or infection of a pathogen. The latter case can be particularly difficult to understand because not only must the proteomes of both the host and the pathogen be understood, but the purpose that pathogen interactions with the host machinery serve must also be understood. What gene expression-based techniques for understanding diseases lack is an establishment of causality, given by mechanistic insight into how these gene expression profiles propagate into disease. The current lack of causality is due to the lack of understanding of the functions encoded by the expressed genes, and if we are to move beyond diagnosis of disease into the development of therapeutics, we need to have this functional insight.

At a more global level, protein function can be interrogated in gene expression datasets through gene set enrichment analysis, which highlights which functions are up and downregulated in a given state relative to baseline expression levels.[3] If this analysis is done with fine enough resolution, one can even begin to assess the contribution each specific gene's expression has to the overall functional state of the cell. However, for the development of therapeutics, one must understand the function of individual proteins in much more detail. If the natural ligand of a protein is known, small molecules which mimic this ligand can be used as competitive inhibitor drugs to modulate the protein's function.[4] Potential drugs can be made with much higher chemical diversity if the structure of the binding pocket is known with high resolution.[5] If the function of a protein involved in disease is conferred through binding to

another protein, peptide and antibody design strategies[6] can be employed to create therapeutic binding partners which target the protein of interest with high specificity, thus lowering the chance of off-target side effects. Both designed peptides/antibodies and small molecule drugs have been employed as therapeutic PPI modulators (inhibitors, stabilizers, etc.),[7,8] but these targets tend to be more difficult to drug than traditional monomeric targets due to factors such as the lack of concavity in most protein interfaces (and thus lack of a clear binding “pocket”), a shortage of high-resolution PPI structures relative to single chain targets, and the potential for protein disorder in the absence of native binding partners. As a result, the targets that are chosen for such studies are restricted not only by the need for their modulation to engender a therapeutic benefit, but also by the ease of designing an effective modulator for a given target. Therefore, the development of therapeutics as well as the expected result of inhibiting the target proteins depend solely on our functional knowledge of proteins as a whole.

## **1.2 Protein structure and function prediction**

It is well known that the tertiary structure of a protein is critical to its function. However, the experimental determination of a protein’s structure remains a cost and time-intensive process, even with recent developments in X-ray crystallography and cryo-EM. Reflective of this difficulty, only ~200,000 protein structures are available in the Protein Data Bank (PDB)[9] despite the hundreds of millions of observed protein sequences. Therefore, in order for protein structures to be determined at a rate rivaling protein sequence acquisition, we must depend on computational methods to model proteins with high enough fidelity that they rival experimental determination.

### ***1.2.1 Protein structure prediction***

The field of protein structure prediction was traditionally divided into two major parts, template-based modeling (TBM) and free modeling (FM). TBM predictors, such as I-TASSER,[10] used threading approaches such as LOMETS,[11] HHsearch,[12] and Raptor,[13] to identify structural templates in the PDB and used these templates to model the query sequence. However, in some cases a suitable template either did not exist or couldn't be found, and therefore FM programs existed to model proteins without templates. However, the most successful FM programs, such as ROSETTA[14] and QUARK,[15] still used previously observed structures in the form of small fragments and knowledge-based potentials. These structure prediction programs were regularly benchmarked in the Critical Assessment of Structure Prediction (CASP) competitions, which have been held every two years since 1994. Among the top-ranked programs at these competitions were Yang Zhang's I-TASSER and David Baker's ROSETTA.

However, the field of protein modeling was revolutionized due to the introduction of coevolution-based deep learning models. Early approaches[16–18] extracted coevolutionary data from multiple sequence alignments (MSAs) to infer inter-residue contacts, which could subsequently be used to improve traditional modeling simulations, particularly in FM programs where no template is provided. However, deep learning architectures became so advanced that in CASP13, DeepMind presented AlphaFold,[19] a program capable of folding a protein solely based on residual convolutional neural network-based inter-residue distance predictions. Distance predictions quickly became popular, with pipelines such as trRosetta[20] and DeepPotential[21] improving upon the original AlphaFold by incorporating orientation and hydrogen bond predictions along with distances. However, this popularity was short-lived because in the following CASP14, DeepMind introduced a novel transformer-based architecture

which many described as having “solved”[22] the structure prediction problem: AlphaFold2.[23] This program was particularly notable in its ability to predict protein structures without the need for intermediate predictions such as contacts or distances, instead being trained from end to end. While Baek et. al.[24] shortly thereafter released a competing program, RoseTTAFold, many still consider AlphaFold2 to be the current state-of-the-art for protein structure prediction, with databases such as Uniprot presenting AlphaFold2 models alongside experimentally solved structures. However, the structure prediction problem is far from solved, as the state-of-the-art predictors still incorrectly predict some structures, particularly full length multi-domain proteins as well as protein-protein complex predictions.[25]

### ***1.2.2 Protein function annotation and prediction***

The space of protein function is diverse and inherently hierarchical based on the detail to which functions are described. In addition, the definition of protein functional annotation is blurred due to distinct yet complimentary sources of functional information. For example, one could experimentally characterize a protein as an enzyme catalyzing metabolite synthesis yet have no notion of the mechanism. On the contrary, many pipelines exist to predict functional features of a protein on a residue level, such as identifying small molecule ligand binding pockets[26] or protein-protein interfaces,[27] yet these programs don’t explicitly provide insight into what function these binding events serve. Despite the difficulty of precisely defining protein function, there exist many classification systems and ontologies which seek to describe these functions in a systematic way. The most popular of these schemas is the Gene Ontology (GO) hierarchy,[28] which annotates protein function through three roots: cellular component (describing the subcellular location of the protein), molecular function (describing the molecular action performed by the protein), and biological process (the larger biological context in which

the protein functions). These GO terms, along with the more specific Human Phenotype Ontology terms,[29] are the central focus of the Critical Assessment of protein Function Annotation algorithms (CAFA) competitions, where function prediction algorithms compete to predict GO terms associated with newly annotated proteins.

According to the most recently released CAFA3 competition,[30] the field of function annotation prediction is not yet dominated by deep learning, with the top-ranking program being based on sequence homology features.[31] Other programs in the field have leveraged structural homology to annotate function, as structure is more conserved than sequence and thus more sensitive to distant evolutionary relationships.[32,33] However, since the last CAFA competition was completed, several deep learning algorithms, including those incorporating state-of-the-art attention mechanisms,[34] have been released. Despite progress in the field, the function prediction problem is far from solved, in part due to the huge number of possible GO terms and the disproportionate amount of sequences annotated with each term (and therefore disproportionate amount of training data available for each term).

### **1.3 Protein-protein interactions**

While some proteins are capable of performing their function as a single chain unit, many proteins form interactions with other proteins in order to perform their function. In fact, a specific subset of proteins (“obligate complexes”) is only properly folded and functional when in complex with another protein. Since some protein functions are only performed in complex with other proteins, to properly characterize protein functions in a proteome-wide fashion, it is required that we first establish the network of protein-protein interactions (PPIs) which occur. This task becomes even more complicated in interspecies systems, such as pathogen-host interactions, where interactions must be considered across divergent proteomes. It should be

noted that PPI networks are not the only functionally relevant protein interaction network (protein-small molecule, protein-nucleic acid, etc.); we choose to focus on PPI networks specifically because of their relationship with protein quaternary structure prediction.

### ***1.3.1 Experimental methods for PPI determination***

Perhaps the most informative and definitive way to establish that two proteins interact is to solve a structure by methods such as cryo-EM and X-ray crystallography which shows the proteins forming an interaction. However, even this method is subject to doubt as the process of preparing the proteins for structure solving can sometimes induce non-biological conformations and interaction behavior. In addition, these methods are relatively low throughput due to the high time and monetary costs associated with them. Another relatively expensive method that provides a higher throughput of interactions at the cost of full structural insight is cross-linking mass spectrometry (XL-MS).[35] Through chemical cross-linking of structurally proximal lysine residues, one can observe physical interactions between proteins, even if those interactions are transient. However, on top of the high effort required in acquiring the crosslinked spectra, the analysis pipelines for this method tend to disagree in their results.[36] Several other experimental binding assays for PPI determination exist, such as isothermal titration calorimetry.[37]

Experimental methods which prioritize coverage of the proteome over precision are typically referred to as high-throughput experimental (HTE) methods. One of these methods is yeast two-hybrid (Y2H),[38] in which the two proteins of interest are fused to two subunits of a transcription factor; if the two proteins interact, the transcription factor induces transcription of a reporter gene. Another popular method is tandem affinity purification (TAP),[39] in which one “bait” protein is tagged with protein domains which allow the bait protein to be purified first on

an IgG column, then a calmodulin column. Any “prey” protein in complex with the bait will also subsequently be purified, and the whole complex is analyzed by mass spectrometry. Many other high-throughput methods exist, such as co-immunoprecipitation,[40] FRET-based screening,[41] proximity biotinylation,[42] etc. However, due to the high throughput of these methods, they are subject to relatively high error rates which cause HTE datasets to frequently be complimentary in coverage rather than overlapping.[43–45]

Unlike protein sequences and structures, there is no singular unified database which contains all experimentally determined protein interactions. Instead, there exist a wide variety of databases which each contain their own subset of protein interactions: IntAct,[46] BioGRID,[47] DIP,[48] etc. In addition, there are several databases which focus solely on specific organisms, such as HAPPI[49] focusing on human protein interactions or VirHostNet[50] presenting all known virus-host protein interactions. In order to make the process of collecting PPI information more convenient, several efforts have been made to compile these databases into meta-databases, such as iRefIndex[51] and IMEx[52]. While there does not exist one dominant resource for PPI data analogous to the PDB for protein structures or Uniprot for protein sequences, many PPI databases can be queried simultaneously using the PSICQUIC service.[53]

### **1.3.2 PPI Prediction**

Since most experimental methods are either too low-throughput or too prone to errors to make reliable whole-proteome PPI networks, many computational biologists have attempted to tackle the problem of PPI prediction. In the early days of the field, PPI prediction was handled through genomic analysis, with features such as genomic proximity indicating a functional association between proteins.[54] However, as HTE and amino acid sequence data became more readily available, approaches dependent on homology between an input and known interactions,

i.e., “interology”, began to appear. Some of these methods, such as PIPE[55] and SPRINT,[56] utilize the huge amount of available sequence data to identify sequence homologous interactions, while others, such as PRISM,[57] iWRAP,[58] Struct2Net,[59] PrePPI,[60] and Threpp[61] utilize much more sensitive structural homology at the cost of data availability. Parallel to homology-based methods, there also exist a wide variety of protein docking methods which assess the likelihood of interaction as a function of structural compatibility, such as HADDOCK,[62] RosettaDock,[63] ZDock,[64] etc., but such methods require that the structure of the individual chains be solved (or modeled with sufficiently high confidence) before predictions can be made.

Like many fields of computational biology, many machine learning-based methods have been developed to solve the problem of PPI prediction. Since early machine learning models were not able to extract information from amino acid sequences directly, the novelty of many early approaches was based in their method of converting sequences into a fixed length vector to be classified: conjoint-triad,[65] autocovariance,[66] etc. As the field of machine learning progressed and deep learning models gained popularity, these algorithmic feature extraction approaches were replaced by deep learning-based approaches which used advanced neural network architectures to embed protein sequences into a fixed length vector. Notable approaches include DPPI,[67] which embedded protein profiles using convolution, PIPR,[68] which embedded sequences using recurrent and convolutional layers, D-SCRIPT,[69] which uses a structure-aware language model to embed sequences and predicts interchain contacts, and RAPPPID,[70] which encodes tokenized protein sequences using an AWD-LSTM embedding. Analogous to docking, there are several deep learning-based pipelines which approach the problem of protein-protein interaction prediction as analogous to the problem of protein complex

structure prediction, assuming that if a reasonable complex structure can be modeled then an interaction must occur. Such predictors include AlphaFold Multimer,[71] a complex-optimized version of AlphaFold2, AF2Complex,[72] an approach which predicts interactions through assessing the interface of AlphaFold2-based complex models, and FoldDock,[73] an approach which combines several types of MSAs (both paired and unpaired) to perform AlphaFold2-based docking. While these approaches can result in high-quality predictions, their computational expense limits them to smaller biological systems, and because of their lack of explicit consideration of non-interacting structures during training of the modelers, these complex predictors may mishandle these cases and try to produce an interface where one shouldn't exist (as is reflected by the fact that AF2Complex saw an improvement over AlphaFold Multimer in part by evaluating structures based on interface score).

While a PPI network can be predicted from scratch using these pairwise approaches, there also exist a class of PPI predictors which use existing network structures to predict new PPIs. While this class of predictor could naively be implemented by drawing edges between proteins with high overlap of interacting partners, many modern predictors use deep learning-based embeddings of the network topology to perform predictions.[74] Particularly noteworthy is the predictor Topsy-Turvy, which integrates both the pairwise prediction of D-SCRIPT and the network-based refinement of GLIDE.[75] However, the existing PPI prediction methods have a few shortcomings. First of all, many predictors don't distinguish between the idea of a functional association between proteins and physical contact. The affirmation of physical contact is necessary for many downstream applications, such as structure-based drug design and protein engineering. Second, many predictors, particularly machine-learning based predictors, are trained and tested on species-specific datasets, yet many biologically important interactions

occur between species, such as host-pathogen interactions during infection. Finally, PPI prediction accuracy faces the same tradeoff between throughput and accuracy that plagues experimental PPI determination. Fast predictors based on sequence searching tend to lack sensitivity, while more accurate deep learning pipelines or structure-based predictors tend to require *a priori* knowledge or a high amount of resources that limit whole-proteome PPI network prediction. As a result, no single predictor has gained popularity over the others because the choice of tool is dependent on what the biological application is.

## **1.4 Other Published Works**

While the primary topic of this dissertation is protein-protein interactions, only the PEPPi manuscript[76] has been published on this topic. Instead, the main body of my published work pertains to protein-ligand analysis and protein structure prediction. With respect to protein-ligand analysis, I have published a book chapter on GPCR virtual screening,[77] a ligand pose distance calculation algorithm,[78] a blind protein-ligand docking algorithm,[79] and a virtual screening of the SurA protein.[80] The majority of my remaining publications deal with our deep learning-based distance/contact map predictors[81–83] and development of I-TASSER.[84–87] Finally, I have participated in a few other various projects such as an analysis of the SARS-CoV-2 proteome[88] and a protein engineering study.[89]

## **1.5 Summary**

Relative to the huge number of known proteins, very few of these proteins are understood in how they function. While these proteins can be experimentally annotated on a case-by-case basis, this is infeasible for a proteome-wide understanding of cells. In addition, proteins do not always function by themselves, and thus annotating them individually misses critical functions

which only occur as part of a protein complex. As a first step to understanding these proteins, we can construct the network of protein interactions present in the cell. While many PPI prediction methods exist, those involving structure are the most promising due to the latent relationship between protein function and protein structure. However, due to the relative lack of available structural information, these PPI predictors must rely on strategies developed in the field of structure prediction to incorporate structural information. In the following chapters, we discuss the development of several structure prediction-inspired PPI classifiers, as well as biological insights we gained from applying these classifiers to specific cross-proteome datasets. First, we discuss a novel PPI prediction method which utilizes homology, functional association, and machine learning to classify protein sequences, and compare this predictor to several competing programs. We then use this PPI predictor to determine the microbe-host interaction network for several probiotic strains of bacteria and analyze this network for novel biological functions. To facilitate analyses of this network, we present a three dimensional network visualization software. Finally, we investigate how interchain coevolution-based PPI predictors could be improved by benchmarking several multiple sequence alignment pairing methods.

## 1.6 References

1. Bateman A, Martin MJ, Orchard S, Magrane M, Agivetova R, Ahmad S, et al. UniProt: The universal protein knowledgebase in 2021. *Nucleic Acids Res.* 2021;49: D480–D489. doi:10.1093/nar/gkaa1100
2. Uffelmann E, Huang QQ, Munung NS, de Vries J, Okada Y, Martin AR, et al. Genome-wide association studies. *Nat Rev Methods Prim.* 2021;1: 1–21. doi:10.1038/s43586-021-00056-9
3. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, et al. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci U S A.* 2005;102: 15545–15550. doi:10.1073/pnas.0506580102
4. Sabe VT, Ntombela T, Jhamba LA, Maguire GEM, Govender T, Naicker T, et al. Current trends in computer aided drug design and a highlight of drugs discovered via computational techniques: A review. *European Journal of Medicinal Chemistry.* Elsevier

- Masson; 2021. p. 113705. doi:10.1016/j.ejmech.2021.113705
5. Liu X, IJzerman AP, van Westen GJP. Computational Approaches for De Novo Drug Design: Past, Present, and Future. *Methods in Molecular Biology*. 2021. pp. 139–165. doi:10.1007/978-1-0716-0826-5\_6
  6. ElGamacy M. Accelerating therapeutic protein design. *Advances in Protein Chemistry and Structural Biology*. Academic Press; 2022. pp. 85–118. doi:10.1016/bs.apcsb.2022.01.004
  7. Lu H, Zhou Q, He J, Jiang Z, Peng C, Tong R, et al. Recent advances in the development of protein–protein interactions modulators: mechanisms and clinical trials. *Signal Transduction and Targeted Therapy*. Nature Publishing Group; 2020. pp. 1–23. doi:10.1038/s41392-020-00315-3
  8. Akhter Y, Hussain R. Protein-protein complexes as targets for drug discovery against infectious diseases. *Adv Protein Chem Struct Biol*. 2020;121: 237–251. doi:10.1016/bs.apcsb.2019.11.012
  9. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, et al. The Protein Data Bank. *Nucleic Acids Res*. 2000;28: 235–242. doi:10.1093/nar/28.1.235
  10. Zhang Y. I-TASSER server for protein 3D structure prediction. *BMC Bioinformatics*. 2008;9: 40. doi:10.1186/1471-2105-9-40
  11. Wu S, Zhang Y. LOMETS: A local meta-threading-server for protein structure prediction. *Nucleic Acids Res*. 2007;35: 3375–3382. doi:10.1093/nar/gkm251
  12. Steinegger M, Meier M, Mirdita M, Vöhringer H, Haunsberger SJ, Söding J. HH-suite3 for fast remote homology detection and deep protein annotation. *BMC Bioinformatics*. 2019;20: 1–15. doi:10.1186/s12859-019-3019-7
  13. Xu J, Li M, Kim D, Xu Y. RAPTOR: optimal protein threading by linear programming. *J Bioinform Comput Biol*. 2003;1: 95–117. doi:10.1142/S0219720003000186
  14. Simons KT, Kooperberg C, Huang E, Baker D. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. *J Mol Biol*. 1997;268: 209–225. doi:10.1006/jmbi.1997.0959
  15. Xu D, Zhang Y. Ab initio protein structure assembly using continuous structure fragments and optimized knowledge-based force field. *Proteins*. 2012;80: 1715–1735. doi:10.1002/prot.24065
  16. Li Y, Hu J, Zhang C, Yu D-J, Zhang Y. ResPRE: high-accuracy protein contact prediction by coupling precision matrix with deep residual neural networks. Valencia A, editor. *Bioinformatics*. 2019 [cited 30 Jul 2019]. doi:10.1093/bioinformatics/btz291
  17. Jones DT, Singh T, Kosciolk T, Tetchner S. MetaPSICOV: Combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. *Bioinformatics*. 2015;31: 999–1006. doi:10.1093/bioinformatics/btu791
  18. Ovchinnikov S, Kamisetty H, Baker D. Robust and accurate prediction of residue-residue interactions across protein interfaces using evolutionary information. *Elife*. 2014;2014. doi:10.7554/eLife.02030
  19. Senior AW, Evans R, Jumper J, Kirkpatrick J, Sifre L, Green T, et al. Improved protein structure prediction using potentials from deep learning. *Nature*. 2020;577: 706–710. doi:10.1038/s41586-019-1923-7
  20. Yang J, Anishchenko I, Park H, Peng Z, Ovchinnikov S, Baker D. Improved protein structure prediction using predicted interresidue orientations. *Proc Natl Acad Sci U S A*. 2020;117: 1496–1503. doi:10.1073/pnas.1914677117
  21. Li Y, Zhang C, Yu DJ, Zhang Y. Deep learning geometrical potential for high-accuracy ab

- initio protein structure prediction. *iScience*. 2022;25: 104425. doi:10.1016/j.isci.2022.104425
22. Callaway E. “It will change everything”: DeepMind’s AI makes gigantic leap in solving protein structures. *Nature*. NLM (Medline); 2020. pp. 203–204. doi:10.1038/d41586-020-03348-4
  23. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Highly accurate protein structure prediction with AlphaFold. *Nature*. 2021;596: 583–589. doi:10.1038/s41586-021-03819-2
  24. Baek M, DiMaio F, Anishchenko I, Dauparas J, Ovchinnikov S, Rie Lee G, et al. Accurate prediction of protein structures and interactions using a three-track neural network. Downloaded from. Nick V Grishin. 2021;10: 22. doi:10.1126/science.abj8754
  25. Pearce R, Zhang Y. Toward the solution of the protein structure prediction problem. *J Biol Chem*. 2021;297: 100870. doi:10.1016/j.jbc.2021.100870
  26. Zhao J, Cao Y, Zhang L. Exploring the computational methods for protein-ligand binding site prediction. *Computational and Structural Biotechnology Journal*. Elsevier; 2020. pp. 417–426. doi:10.1016/j.csbj.2020.02.008
  27. Xue LC, Dobbs D, Bonvin AMJJ, Honavar V. Computational prediction of protein interfaces: A review of data driven methods. *FEBS Letters*. No longer published by Elsevier; 2015. pp. 3516–3526. doi:10.1016/j.febslet.2015.10.003
  28. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, et al. Gene ontology: Tool for the unification of biology. *Nature Genetics*. Nature Publishing Group; 2000. pp. 25–29. doi:10.1038/75556
  29. Köhler S, Gargano M, Matentzoglou N, Carmody LC, Lewis-Smith D, Vasilevsky NA, et al. The human phenotype ontology in 2021. *Nucleic Acids Res*. 2021;49: D1207–D1217. doi:10.1093/nar/gkaa1043
  30. Zhou N, Jiang Y, Bergquist TR, Lee AJ, Kacsoh BZ, Crocker AW, et al. The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biol*. 2019;20: 1–23. doi:10.1186/s13059-019-1835-8
  31. You R, Zhang Z, Xiong Y, Sun F, Mamitsuka H, Zhu S. GOLabeler: Improving sequence-based large-scale protein function prediction by learning to rank. *Bioinformatics*. 2018;34: 2465–2473. doi:10.1093/bioinformatics/bty130
  32. Zhang C, Zheng W, Freddolino PL, Zhang Y. MetaGO: Predicting Gene Ontology of Non-homologous Proteins Through Low-Resolution Protein Structure Prediction and Protein–Protein Network Mapping. *J Mol Biol*. 2018;430: 2256–2265. doi:10.1016/j.jmb.2018.03.004
  33. Zhang C, Freddolino PL, Zhang Y. COFACTOR: Improved protein function prediction by combining structure, sequence and protein-protein interaction information. *Nucleic Acids Res*. 2017;45: W291–W299. doi:10.1093/nar/gkx366
  34. Lai B, Xu J. Accurate protein function prediction via graph attention networks with predicted structure information. *Brief Bioinform*. 2022;23. doi:10.1093/bib/bbab502
  35. Mintseris J, Gygi SP. High-density chemical cross-linking for modeling protein interactions. *Proc Natl Acad Sci U S A*. 2020;117: 93–102. doi:10.1073/pnas.1902931116
  36. Yilmaz Ş, Busch F, Nagaraj N, Cox J. Accurate and Automated High-Coverage Identification of Chemically Cross-Linked Peptides with MaxLynx. *Anal Chem*. 2022;94: 1608–1617. doi:10.1021/acs.analchem.1c03688

37. Velazquez-Campoy A, Leavitt SA, Freire E. Characterization of protein-protein interactions by isothermal titration calorimetry. *Methods Mol Biol.* 2004;261: 35–54. doi:10.1385/1-59259-762-9:035
38. Fields S, Song OK. A novel genetic system to detect protein-protein interactions. *Nature.* 1989;340: 245–246. doi:10.1038/340245a0
39. Puig O, Caspary F, Rigaut G, Rutz B, Bouveret E, Bragado-Nilsson E, et al. The tandem affinity purification (TAP) method: A general procedure of protein complex purification. *Methods.* 2001;24: 218–229. doi:10.1006/meth.2001.1183
40. Mann M, Hendrickson RC, Pandey A. Analysis of Proteins and Proteomes By Mass Spectrometry. *Annu Rev Biochem.* 2001;70: 437–473. Available: <http://www.ncbi.nlm.nih.gov/pubmed/11395414>
41. Margineanu A, Chan JJ, Kelly DJ, Warren SC, Flatters D, Kumar S, et al. Screening for protein-protein interactions using Förster resonance energy transfer (FRET) and fluorescence lifetime imaging microscopy (FLIM). *Sci Rep.* 2016;6: 1–17. doi:10.1038/srep28186
42. Fernández-Suárez M, Chen TS, Ting AY. Protein-protein interaction detection in vitro and in cells by proximity biotinylation. *J Am Chem Soc.* 2008;130: 9251–9253. doi:10.1021/ja801445p
43. Fields S. High-throughput two-hybrid analysis: The promise and the peril. *FEBS J.* 2005;272: 5391–5399. doi:10.1111/j.1742-4658.2005.04973.x
44. Sprinzak E, Sattath S, Margalit H. How reliable are experimental protein-protein interaction data? *J Mol Biol.* 2003;327: 919–923. doi:10.1016/S0022-2836(03)00239-0
45. Gentleman R, Huber W. Making the most of high-throughput protein-interaction data. *Genome Biol.* 2007;8: 1–10. doi:10.1186/gb-2007-8-10-112
46. Orchard S, Ammari M, Aranda B, Breuza L, Briganti L, Broackes-Carter F, et al. The MIntAct project - IntAct as a common curation platform for 11 molecular interaction databases. *Nucleic Acids Res.* 2014;42: D358–D363. doi:10.1093/nar/gkt1115
47. Oughtred R, Stark C, Breitkreutz BJ, Rust J, Boucher L, Chang C, et al. The BioGRID interaction database: 2019 update. *Nucleic Acids Res.* 2019;47: D529–D541. doi:10.1093/nar/gky1079
48. Salwinski L. The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res.* 2004;32: 449D – 451. doi:10.1093/nar/gkh086
49. Chen JY, Pandey R, Nguyen TM. HAPPI-2: A Comprehensive and High-quality Map of Human Annotated and Predicted Protein Interactions. *BMC Genomics.* 2017;18: 1–16. doi:10.1186/s12864-017-3512-1
50. Guirimand T, Delmotte S, Navratil V. VirHostNet 2.0: Surfing on the web of virus/host molecular interactions data. *Nucleic Acids Res.* 2015;43: D583–D587. doi:10.1093/nar/gku1121
51. Razick S, Magklaras G, Donaldson IM. iRefIndex: A consolidated protein interaction database with provenance. *BMC Bioinformatics.* 2008;9: 1–19. doi:10.1186/1471-2105-9-405
52. Orchard S, Kerrien S, Abbani S, Aranda B, Bhate J, Bidwell S, et al. Protein interaction data curation: The International Molecular Exchange (IMEx) consortium. *Nature Methods.* Nature Publishing Group; 2012. pp. 345–350. doi:10.1038/nmeth.1931
53. Aranda B, Blankenburg H, Kerrien S, Brinkman FSL, Ceol A, Chautard E, et al. PSICQUIC and PSISCORE: Accessing and scoring molecular interactions. *Nature*

- Methods. Nature Publishing Group; 2011. pp. 528–529. doi:10.1038/nmeth.1637
54. Snel B, Lehmann G, Bork P, Huynen MA. STRING: a web-server to retrieve and display the repeatedly occurring neighbourhood of a gene. *Nucleic Acids Res.* 2000;28: 3442–4. Available: <http://www.ncbi.nlm.nih.gov/pubmed/10982861>
  55. Pitre S, Dehne F, Chan A, Cheetham J, Duong A, Emili A, et al. PIPE: A protein-protein interaction prediction engine based on the re-occurring short polypeptide sequences between known interacting protein pairs. *BMC Bioinformatics.* 2006;7: 1–15. doi:10.1186/1471-2105-7-365
  56. Li Y, Ilie L. SPRINT: Ultrafast protein-protein interaction prediction of the entire human interactome. *BMC Bioinformatics.* 2017;18: 1–11. doi:10.1186/s12859-017-1871-x
  57. Baspinar A, Cukuroglu E, Nussinov R, Keskin O, Gursoy A. PRISM: A web server and repository for prediction of protein-protein interactions and modeling their 3D complexes. *Nucleic Acids Res.* 2014;42: W285–W289. doi:10.1093/nar/gku397
  58. Hosur R, Xu J, Bienkowska J, Berger B. iWRAP: An Interface Threading Approach with Application to Prediction of Cancer-Related Protein–Protein Interactions. *J Mol Biol.* 2011;405: 1295–1310. doi:10.1016/J.JMB.2010.11.025
  59. Singh R, Xu J, Berger B. Struct2Net: Integrating Structure into Protein-Protein Interaction Prediction. *Pacific Symp Biocomput.* 2006. Available: <http://theory.csail.mit.edu/struct2net>
  60. Zhang QC, Petrey D, Deng L, Qiang L, Shi Y, Thu CA, et al. Structure-based prediction of protein-protein interactions on a genome-wide scale. *Nature.* 2012;490: 556–560. doi:10.1038/nature11503
  61. Gong W, Guerler A, Zhang C, Warner E, Li C, Zhang Y. Integrating Multimeric Threading With High-throughput Experiments for Structural Interactome of *Escherichia coli*. *J Mol Biol.* 2021;433: 166944. doi:10.1016/j.jmb.2021.166944
  62. Dominguez C, Boelens R, Bonvin AMJJ. HADDOCK: A protein-protein docking approach based on biochemical or biophysical information. *J Am Chem Soc.* 2003;125: 1731–1737. doi:10.1021/ja026939x
  63. Gray JJ, Moughon S, Wang C, Schueler-Furman O, Kuhlman B, Rohl CA, et al. Protein-protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations. *J Mol Biol.* 2003;331: 281–299. doi:10.1016/S0022-2836(03)00670-3
  64. Pierce BG, Wiehe K, Hwang H, Kim BH, Vreven T, Weng Z. ZDOCK server: Interactive docking prediction of protein-protein complexes and symmetric multimers. *Bioinformatics.* 2014;30: 1771–1773. doi:10.1093/bioinformatics/btu097
  65. Shen J, Zhang J, Luo X, Zhu W, Yu K, Chen K, et al. Predicting protein-protein interactions based only on sequences information. *Proc Natl Acad Sci U S A.* 2007;104: 4337–4341. doi:10.1073/pnas.0607879104
  66. Guo Y, Yu L, Wen Z, Li M. Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences. *Nucleic Acids Res.* 2008;36: 3025–3030. doi:10.1093/nar/gkn159
  67. Hashemifar S, Neyshabur B, Khan AA, Xu J. Predicting protein-protein interactions through sequence-based deep learning. *Bioinformatics.* 2018;34: i802–i810. doi:10.1093/bioinformatics/bty573
  68. Chen M, Ju CJT, Zhou G, Chen X, Zhang T, Chang KW, et al. Multifaceted protein-protein interaction prediction based on Siamese residual RCNN. *Bioinformatics.* 2019;35: i305–i314. doi:10.1093/bioinformatics/btz328

69. Sledzieski S, Singh R, Cowen L, Berger B. D-SCRIPT translates genome to phenome with sequence-based, structure-aware, genome-scale predictions of protein-protein interactions. *Cell Syst.* 2021;12: 969-982.e6. doi:10.1016/j.cels.2021.08.010
70. Szymborski J, Emad A. RAPPID: towards generalizable protein interaction prediction with AWD-LSTM twin networks. *Bioinformatics.* 2022;38: 3958–3967. doi:10.1093/bioinformatics/btac429
71. Evans R, O'Neill M, Pritzel A, Antropova N, Senior A, Green T, et al. Protein complex prediction with AlphaFold-Multimer. *bioRxiv.* 2021; 2021.10.04.463034. doi:10.1101/2021.10.04.463034
72. Gao M, Nakajima An D, Parks JM, Skolnick J. AF2Complex predicts direct physical interactions in multimeric proteins with deep learning. *Nat Commun.* 2022;13: 1–13. doi:10.1038/s41467-022-29394-2
73. Bryant P, Pozzati G, Elofsson A. Improved prediction of protein-protein interactions using AlphaFold2. *Nat Commun.* 2022;13: 1–11. doi:10.1038/s41467-022-28865-w
74. Coşkun M, Koyutürk M. Node similarity-based graph convolution for link prediction in biological networks. *Bioinformatics.* 2021;37: 4501–4508. doi:10.1093/bioinformatics/btab464
75. Singh R, Devkota K, Sledzieski S, Berger B, Cowen L. Topsy-Turvy: Integrating a global view into sequence-based PPI prediction. *Bioinformatics.* 2022;38: I264–I272. doi:10.1093/bioinformatics/btac258
76. Bell EW, Schwartz JH, Freddolino PL, Zhang Y. PEPPI: Whole-proteome Protein-protein Interaction Prediction through Structure and Sequence Similarity, Functional Association, and Machine Learning. *J Mol Biol.* 2022; 167530. doi:10.1016/j.jmb.2022.167530
77. Chan W, Wu J, Bell E, Zhang Y. Virtual Screening and Bioactivity Modeling for G Protein-Coupled Receptors. *GPCRs as Therapeutic Targets.* John Wiley & Sons, Ltd; 2022. pp. 388–423. doi:10.1002/9781119564782.ch12
78. Bell EW, Zhang Y. DockRMSD: an open-source tool for atom mapping and RMSD calculation of symmetric molecules through graph isomorphism. *J Cheminform.* 2019;11: 40. doi:10.1186/s13321-019-0362-7
79. Zhang W, Bell EW, Yin M, Zhang Y. EDock: Blind protein-ligand docking by replica-exchange monte carlo simulation. *J Cheminform.* 2020;12: 1–17. doi:10.1186/s13321-020-00440-9
80. Bell EW, Zheng EJ, Ryno LM. Identification of inhibitors of the E. coli chaperone SurA using in silico and in vitro techniques. *Bioorganic Med Chem Lett.* 2018;28: 3540–3548. doi:10.1016/j.bmcl.2018.09.034
81. Li Y, Zhang C, Zheng W, Zhou X, Bell EW, Yu DJ, et al. Protein inter-residue contact and distance prediction by coupling complementary coevolution features with deep residual networks in CASP14. *Proteins.* 2021;89: 1911–1921. doi:10.1002/prot.26211
82. Li Y, Zhang C, Bell EW, Zheng W, Zhou X, Yu D-J, et al. Deducing high-accuracy protein contact-maps from a triplet of coevolutionary matrices through deep residual convolutional networks. Kolodny R, editor. *PLOS Comput Biol.* 2021;17: e1008865. doi:10.1371/journal.pcbi.1008865
83. Li Y, Zhang C, Bell EW, Yu DJ, Zhang Y. Ensembling multiple raw coevolutionary features with deep residual neural networks for contact-map prediction in CASP13. *Proteins.* 2019;87: 1082–1091. doi:10.1002/prot.25798
84. Zhou X, Zheng W, Li Y, Pearce R, Zhang C, Bell EW, et al. I-TASSER-MTD: a deep-

- learning-based platform for multi-domain protein structure and function prediction. *Nat Protoc.* 2022;17: 2326–2353. doi:10.1038/s41596-022-00728-0
85. Zheng W, Li Y, Zhang C, Zhou X, Pearce R, Bell EW, et al. Protein structure prediction using deep learning distance and hydrogen-bonding restraints in CASP14. *Proteins.* 2021;89: 1734–1751. doi:10.1002/prot.26193
  86. Zheng W, Zhang C, Li Y, Pearce R, Bell EW, Zhang Y. Folding non-homologous proteins by coupling deep-learning contact maps with I-TASSER assembly simulations. *Cell Reports Methods.* 2021;1: 100014. doi:10.1016/j.crmeth.2021.100014
  87. Zheng W, Zhang C, Bell EW, Zhang Y. I-TASSER gateway: A protein structure and function prediction server powered by XSEDE. *Futur Gener Comput Syst.* 2019;99: 73–85. doi:10.1016/j.future.2019.04.011
  88. Zhang C, Zheng W, Huang X, Bell EW, Zhou X, Zhang Y. Protein Structure and Sequence Reanalysis of 2019-nCoV Genome Refutes Snakes as Its Intermediate Host and the Unique Similarity between Its Spike Protein Insertions and HIV-1. *J Proteome Res.* 2020;19: 1351–1360. doi:10.1021/acs.jproteome.0c00129
  89. Li H, Yi S, Bell EW, Huang Q, Zhao X. Recombinant *Penicillium oxalicum* 16  $\beta$ -Glucosidase 1 Displays Comprehensive Inhibitory Resistance to Several Lignocellulose Pretreatment Products, Ethanol, and Salt. *Appl Biochem Biotechnol.* 2020;191: 772–784. doi:10.1007/s12010-019-03183-y

## **Chapter 2 PEPPI: Whole-proteome Protein-protein Interaction Prediction through Structure and Sequence Similarity, Functional Association, and Machine Learning**

This chapter was adapted from the manuscript Eric W. Bell, Jacob H. Schwartz, Peter L. Freddolino, Yang Zhang. “PEPPI: Whole-proteome Protein-protein Interaction Prediction through Structure and Sequence Similarity, Functional Association, and Machine Learning” (2022). Journal of Molecular Biology 434(11): 167530.[1]

### **2.1 Abstract**

Proteome-wide identification of protein-protein interactions is a formidable task which has yet to be sufficiently addressed by experimental methodologies. Many computational methods have been developed to predict proteome-wide interaction networks, but few leverage both the sensitivity of structural information and the wide availability of sequence data. We present PEPPI, a pipeline which integrates structural similarity, sequence similarity, functional association data, and machine learning-based classification through a naïve Bayesian classifier model to accurately predict protein-protein interactions at a proteomic scale. Through benchmarking against a set of 798 ground truth interactions and an equal number of non-interactions, we have found that PEPPI attains 4.5% higher AUROC than the best of other state-of-the-art methods. As a proteomic-scale application, PEPPI was applied to model the interactions which occur between SARS-CoV-2 and human host cells during coronavirus infection, where 403 interactions were identified with predictions covering 73% of a gold standard dataset from PSICQUIC and demonstrating significant complementarity with the most

recent high-throughput experiments. PEPPI is available both as a webserver and in a standalone version and should be a powerful and generally applicable tool for computational screening of protein-protein interactions.

## **2.2 Introduction**

The biological function of many proteins is conferred through their interactions with other proteins. Therefore, to fully understand the function of each protein in an organism, one must first attain a comprehensive network of the protein-protein interactions (PPIs) that occur within the cell. The discovery of critical interactions within this interaction network, or “interactome”, can lead to drug development[2] or protein engineering[3,4] targeting these interactions. However, many of these experiments do not guarantee that the interactions detected are, in fact, direct physical contacts between the proteins; some of the earliest databases for PPI prediction involve features that assert only a functional association between proteins.[5] While these databases can be useful for prediction of physical interactions, as all physical interactions are functionally associated, the converse is not true; many biological applications, such as drug target discovery, require knowledge of which proteins come into physical contact. The methods for elucidating these direct physical interactions are at present either prohibitively costly for whole-proteome analysis (such as structure solving or crosslinking mass spectrometry) or are too susceptible to errors (such as yeast-two hybrid).[6] As an alternative, computational methods can be used to model proteome-wide interactions, as well as refine existing interaction datasets.

One of the most straightforward methods of computational interaction prediction is to determine whether the query protein pair is similar to an already known interaction. Many programs directly leverage sequence similarity for this purpose because the sequence comparison operation is quick and sequence data is plentiful.[7,8] However, since structure is more

evolutionarily conserved than sequence, structural similarity is much more effective at detecting distantly similar PPIs; methods which leverage this structural information,[9–11] in particular the method PrePPI which combines homology modeling-based structural features with functional association using a Bayesian network consensus,[12] grow more powerful as modern structural biology methods such as cryo-EM facilitate the solving of complicated protein complex structures and as computational approaches offer improved accuracy in predicting the folds of individual proteins.[13] In addition, structures provide a clear ground truth as to whether two proteins interact physically; if a solved structure of the interaction exists, the proteins are likely to interact *in vivo*. Therefore, an effective similarity-based program should consider both structural and sequence similarity.

Another common method for PPI prediction is the application of machine learning-based classifiers. In the earlier days of machine learning, the major novelty of machine learning-based PPI predictors was in how they extracted features from the input amino acid sequences in order to create a fixed-length vector that could be utilized in standard machine learning algorithms, such as the conjoint triad method[14] or autocorrelation.[15] As more modern deep learning techniques became available, improved PPI predictions were achieved with features solely from evolutionary profiles[16] or even based on sequence alone.[17,18] Despite these initial studies into deep learning, to our knowledge these methods have yet to demonstrate success in a species-agnostic context; more commonly, they are benchmarked by either combining a few species-specific datasets[17] or by training on one species and testing on another.[16,18]

Here, we present a Pipeline for the Extraction of Predicted Protein-protein Interactions (PEPPI), which offers high-accuracy PPI predictions through a consensus of sequence and structural similarity, functional association, and neural network classification. While the source

code for PEPPI can be found at <https://github.com/ewbell94/PEPPI>, an online webserver implementation of this pipeline can be found at <https://zhanggroup.org/PEPPI/>, which allows users to create PPI predictions from sequences alone. We additionally present an application of PEPPI to make predictions of the inter-species interactome between human host cells and SARS-CoV-2. Through the following benchmarks and examples, we demonstrate that PEPPI is a useful tool for predicting both pairwise and systems-level PPIs.

## 2.3 Results

### 2.3.1 Pipeline overview and module cross-validation

PEPPI is a protein-protein interaction prediction pipeline which takes in a pair of query sequences and quantifies their likelihood of interaction as a natural log-transformed likelihood ratio ( $\log(\text{LR})$ ) through a consensus of five independent prediction modules (**Figure 2.1**). This consensus is determined by a naïve Bayesian classifier model trained on a set of 800 high-confidence interactions from IntAct[19] and 800 curated non-interactions from the Negatome 2.0 database[20] (see Methods).

**Figure 2.2a** presents the 10-fold cross validation performance of each individual module and the full pipeline on this training set. The best-performing individual modules are SPRING and SEQ, which implement structure and sequence-based similarity approaches, respectively. The SPRING module uses the dimeric threading program SPRING[21] to identify dimer structure templates out of a database of interacting proteins extracted from the PDB, while SEQ uses BLAST[22] sequence searching to identify similar interactions in a database of direct interactions identified by high-throughput experiment (HTE) data. These homology-based modules will perform well for any cases which have homologous similarity to existing interactions, which is the case for many true interactions. The next best-performing module is the

neural network-based CT module, which transforms the input amino acid sequences into a fixed-length vector according to the conjoint triad method[14] and classifies the resulting vector through a neural network model. This module helps PEPPI retrieve true positive predictions in case there is only loose homologous similarity to existing interactions. The STRING module, a module which extracts various query functional association features from the STRING database,[23] performs relatively poorly on its own. This is expected because of its focus on functional association data instead of physical interaction data and because of its inability to provide data if the interaction is not located in STRING. SPRINGNEG has a nearly identical methodology to SPRING but it performs least well because it searches through a database of non-interacting protein structures. Therefore, a hit in this database will only lower the interaction score because by design it only provides information to filter out functionally associated non-interactions. Overall, the combination of all modules clearly outperforms any single module, demonstrating that the modules are complementary in classification. This combination is valuable because it prevents any one module from dominating the final classification; any individual module's positive classification can be overcome by sufficiently confident negative classification by the other modules and vice versa.

### ***2.3.2 PEPPI benchmark and performance***

In order to quantify the classification performance of PEPPI against existing methods, we benchmarked PEPPI alongside PRISM,[9] a structure-based similarity predictor, SPRINT,[7] a sequence-based similarity predictor, PIPR,[17] a deep learning predictor which utilizes a combination of recurrent and convolutional layers in its architecture, and D-SCRIPT,[18] a “structure aware” deep learning predictor improving upon PIPR. These benchmarks were performed using a randomly selected test set of 798 interacting structure pairs and an equal

number of structure pairs involving chains from the same protein complex but known to not come into physical contact (and thus do not form a physical interaction). All structures of this test set were taken from the PDB and were confirmed to have < 50% sequence identity to the PEPPI training protein pairs. It should be noted that the protein pairs in our dataset are classified through their sequence alone, and therefore, even though many of the corresponding PDB structures contain more than one interaction, each pair of chains can be considered independently.

The results of the PPI predictions are summarized presented in **Figure 2.2b** and **Table 2.1**, where it is shown that PEPPI significantly outperforms all competing methods in terms of area under ROC (AUROC), average precision (similar to area under precision-recall curve, AUPRC), and all but SPRINT in maximum achievable Matthews correlation coefficient (MCC). PRISM had the lowest performance in this benchmark, which is likely due to its outdated interface structure library, which misses many structures which have been solved since its release to the public. Interestingly, the highest performance from a competing program was seen from SPRINT, a sequence motif-based similarity classifier, and not from either D-SCRIPT or PIPR, the more sophisticated deep-learning pipelines. The deep learning architectures of PIPR and D-SCRIPT were originally trained and benchmarked based on performance of species-specific interaction datasets; when these methods were applied to the species agnostic dataset used in our benchmarks, the deep learning methods' ability to accurately classify interactions decreased. As a result, the competing method which draws its conclusions from explicit similarity (SPRINT) outperforms the methods which try to sub-optimally learn the interaction problem itself (PIPR & D-SCRIPT). However, we still found that D-SCRIPT outperforms PIPR, as is consistent with D-SCRIPT's benchmarks.[18]

One particular case of interest in our benchmark dataset is the interaction between chain A and chain B of PDB code 1F3M (corresponding to the N- and C-terminal domains of the human kinase PAK1). While this is a true interaction (and PEPPI correctly classifies it as such), no hit was found by PEPPI in either the sequence or structure databases after homologous template removal, leading to poor scores for those pipelines (SPRING: 8.937, STRING: not found, SEQ: 0.129, SPRINGNEG: 4.618). However, this interaction was still classified as positive with a  $\log(\text{LR})$  of 0.056 due to a high interaction probability from the CT module (0.999), thus demonstrating the utility of CT for rescuing interactions that do not attain significant similarity. On the side of non-interaction classification, the potential interaction of chain G and chain I of 3CJH (corresponding to two Tim13 chains of the yeast Tim8-Tim13 complex) poses an interesting case, as these chains are present in the same complex but do not directly interact with each other. PEPPI was able to classify this as non-interacting ( $\log(\text{LR}) = -0.939$ ) where competing programs could not, despite attaining a high SPRING score (34.208), a high CT confidence (1.0), and loose SEQ homology (0.217). The reason for the correct classification is a high score from SPRINGNEG (36.108) which pushes down the total interaction likelihood, thus demonstrating a case where SPRINGNEG's false positive identification ability rescues the pipeline from misclassifying the interaction.

While the previous balanced dataset is convenient for benchmarking, it is not fully reflective of the context in which an interactome prediction algorithm is applied because true interactions are much sparser relative to the total set of pairwise combinations of query proteins in almost all contexts. Therefore, we randomly sampled 100 interacting pairs from the previous test set and paired the 200 chains from these interactions in an all-by-all fashion (excluding self pairings), resulting in an unbalanced test set of 100 true interacting pairs and 19,890 decoy pairs.

These decoys are not confirmed non-interactions, but due to the low likelihood of finding interactions by random chance, benchmarking using this dataset is still valuable. Due to the high number of protein pairs, PRISM was excluded from this benchmark because of its slow speed and poor performance on the preceding benchmark. The results of this unbalanced benchmark are presented in **Figure 2.2** and **Table 2.2**. The outcome is similar to the previous benchmark, with PEPPI outperforming all other programs, followed most closely by the sequence-based algorithm SPRINT. In this benchmark, however, the superiority of PEPPI is much clearer, as SPRINT is on average more susceptible to false positive detection for comparable recalls. This resulted in a statistically significant difference in max MCC performance between PEPPI and SPRINT, which was not the case on the balanced benchmark set. Also made clearer is the relatively poor performance of PIPR with respect to false positive errors; so many pairs are classified with the highest confidence score that the maximum achievable precision is 0.186. D-SCRIPT again clearly outperforms PIPR but fails to perform as well as the similarity-based methods. One particular case of interest in this benchmark is the true interaction between chain J and chain K of PDB code 3CI0 (part of the type 2 secretion system of enterotoxigenic *E. coli*), an interaction which was detected only by PEPPI (**Figure 2.2d**). This interaction was able to be detected solely through structural similarity (SPRING: 51.45), with all other modules failing to detect the interaction (STRING: not found, SPRINGNEG: 4.09, SEQ: 0.279, CT: 0.026), leading to a log(LR) of 1.433.

### **2.3.3 SARS-CoV-2 and human interactome modeling**

The COVID-19 pandemic, caused by the SARS-CoV-2 virus, has disrupted the lives of almost every person to some degree, and as of November 2021, over 5 million people have lost their lives to the disease worldwide.[24] SARS-CoV-2 has thus become an essential entity to

understand, as our expedient comprehension of this virus translates to the development of therapeutic medicines, such as antiviral drugs and vaccines, for current and future coronavirus infections. One fundamental step towards understanding the function of the virus is to model the virus-host interactome, with which we can begin to identify the purpose of each viral protein through our functional understanding of the human proteins with which each viral protein interacts. To this end, we have predicted the set of interactions which occur between SARS-CoV-2 and human proteins using PEPPI.

Our SARS-CoV-2/Human interactome model consists of 403 interactions whose likelihood ratios were determined to be greater than 1, i.e., interactions which are more likely to be interacting than not. As shown in **Figure 2.3a-b**, the SARS-CoV-2 protein which has the highest number of predicted interactions is the Spike protein (86 interactions), followed by the 2'-O-methyltransferase nsp16 (46 interactions), and the RNA polymerase nsp12 (41 interactions). The highest confidence interaction of this network was the Spike/ACE2 interaction (**Figure 2.3c**), which is expected given the extensive study of this interaction due to its essential role in viral entry.[25] PEPPI also correctly predicted Spike to interact with two other host proteins important to viral entry: Furin, a protease which cleaves Spike during entry of SARS-CoV-2 but is not involved in SARS-CoV-1 entry,[26] and TMPRSS2, a cell-surface protease involved in viral entry of both SARS-CoV-2 and SARS-CoV-1.[25] The PEPPI results also demonstrated the power of structure similarity-based PPI prediction through the prediction of the PARP15-nsp3 interaction (**Figure 2.3d**); this interaction was predicted with high confidence ( $\log(\text{LR}) = 1.435$ ), mainly due to the SPRING module's high confidence score (35.6). The PARP proteins are known to interact with the nsp3 macrodomain in other coronaviruses,[27] so detection of this interaction in our dataset stands as an important validation.

To evaluate PEPPI's overall performance at recapitulating known biology, we constructed a gold standard dataset of known interactions between human and SARS coronavirus proteins from PSICQUIC[28] for comparison. This dataset consisted of 128 interactions, 94 (73%) of which were predicted by PEPPI. In addition, we compared the overlap between our predicted dataset and a recently published high-throughput experimental dataset.[29] PEPPI's predictions only shared one interaction with this dataset (an interaction between MARK3 and ORF9b), but the experimental dataset presents only functional associations due to their use of AP-MS, which is known to pull down entire interacting complexes instead of only the “prey” protein of interest. Compared to PEPPI, the experimental dataset also misses crucial interactions, such as the interactions involving the Spike protein with ACE2, Furin, and TMPRSS2. In fact, only 3 of the 128 (2%) gold standard interactions we isolated from PSICQUIC are present in this dataset. Thus, even in the presence of a high throughput experimental dataset, PEPPI provides a demonstrably useful complement and reveals many direct physical interactions which would otherwise be missed.

Lastly, PEPPI made the potentially significant predictions that nsp3 interacts with the post-translational modifiers NEDD8 and UBD (FAT10). While it is well-documented that the papain-like protease (PLPro) of SARS-CoV-2 nsp3 both deISGylates and deubiquitinates viral proteins to avoid host detection and thus evade immune response,[30] there has been less study of the role of the related, ubiquitin-like, post-transcriptional modifiers NEDD8 and UBD in SARS-CoV-2 disease. NEDD8 tags proteins for degradation, has been implicated in the innate immune response to viruses,[31] and is a target by some viruses for modulation of host immune response.[32] UBD has been shown to be a ubiquitin-independent and cytokine-inducible modifier which targets proteins for proteasomal degradation[33] and has additionally been

shown to have roles in viral infection defense.[34] Furthermore, the top structural templates PEPPI found for nsp3/NEDD8 (**Figure 2.3e**) and nsp3/UBD (**Figure 2.3f**) were a PLPro in complex with free ubiquitin and a PLPro in complex with the ubiquitin-like and innate-immune-modulating protein ISG15, respectively. We therefore hypothesize that SARS-CoV-2 modulates host innate immune response through interaction of nsp3 with NEDD8 and UBD in a similar manner that nsp3 interacts with ubiquitin.

## 2.4 Discussion & conclusion

We have presented a novel PPI prediction pipeline which demonstrates superior performance relative to other approaches. In addition to performance, this method presents a few unique advantages. Firstly, because the structure-based analysis makes use of threading rather than structural alignment, it is much faster than pipelines which need to explicitly model the input chains, while retaining the flexibility of not requiring an input structure. Second, because structure-based analysis is a component of the pipeline, PEPPI can produce rough structural models of the interactions, which can help deepen biological insights such as interface residue determination and can guide follow-up experiments. Finally, because PEPPI is a consensus model, it is not solely dependent on any one methodology to make its predictions; even if all modules classify an interaction with low confidence, if these classifications agree, the final prediction will have reasonable confidence (as we have shown in a previous example above). In addition, the consensus classifier is constructed such that if any modules are intentionally excluded or fail to produce a score, a prediction can still be made from the remaining modules.

A few shortcomings and assumptions of the pipeline should also be discussed. First, the interaction predictions made in this pipeline are based largely on similarity to known PPIs, and these modules will only detect interactions with similarity to solved structures or to interactions

detected in high-throughput screens. Therefore, the method's performance will depend on the coverage of our knowledge of the existing interaction space, which is currently far from fully comprehensive. However, this knowledge will expand as more interactions are discovered, so the power of the similarity-based method will improve over time. Second, because PEPPI is similarity-dependent, if an interaction is predicted between proteins of two given families, all other proteins in those families will likely also be predicted to interact. In this case, interactions involving proteins of the same families can be sorted by LR; the highest rated interaction is the most likely to be true. Finally, this pipeline predicts the capability for proteins to interact regardless of biological context. As a result, it is possible for some of the interactions predicted here to not exist within the context of the cell due to factors such as incompatible subcellular localization or insufficient expression of the proteins of interest *in vivo*. While this additional biological insight can be useful in pruning the interaction space in a proteome-wide interactome modeling study, it is not explicitly considered in PEPPI's interaction predictions. Therefore, it is worth validating the interactions predicted with this program with more focused small-scale biochemical studies, such as crosslinking mass spectrometry experiments. Despite these shortcomings, the whole proteome interaction networks modeled by PEPPI can help biologists retrieve existing biology and derive novel biology for their system of interest, as we did for the SARS-CoV-2/Human interaction system. Through the understanding of PPI networks on the whole-proteome scale that PEPPI provides, future studies will be able to better understand the systems-level complexity that underpins biological phenomena as well as target individual edges of the network for therapeutic benefit.

## **2.5 Methods**

### ***2.5.1 PPI prediction by SPRING threading ("SPRING"/"SPRINGNEG" module)***

SPRING[21] is a program which detects a template dimer structure given a pair of query sequences. However, this program has been altered from its original published state in order to perform better on the task of PPI classification, and therefore, we summarize the new procedure here. First, the pair of query sequences are threaded monomerically by HHsearch[35] through a combination dimer and non-interaction structure library in order to identify monomer structure templates which fit each respective query sequence. Each template structure is scored by the raw alignment score for each query-template alignment; these scores are then Z-normalized by the mean and standard deviation of the top 20,000 templates. A similarity model is constructed for each query chain according to the highest scoring query-template alignment. The top 5,000 template structures (by Z-score) for each of the two chains are paired all-by-all in search of an identical pair of chain IDs in the dimer structure library for SPRING (or the negative structure library for SPRINGNEG), where each identified dimer template pair is scored by the minimum of the two chains' respective monomeric Z-scores. Once dimer template pairs have been identified, a dimeric model is constructed for the top 100 dimer template pairs by TM-align[36] superposition of the similarity model of each chain onto the dimeric template structure. Each model is then scored according to the SPRINGscore:

$$SPRINGscore = Z + w_1 TM + w_2 Dcomplex \quad (1)$$

where  $Z$  is the Z-score of the dimer template,  $TM$  is the minimum structure similarity from superposition,  $Dcomplex$  is the model interface energy as calculated by a modified form of DCOMPLEX[37] which only considers  $C\alpha$  atoms, and  $w_1$  and  $w_2$  are weights with value 4.0 a-d - 0.1 (negative because a lower DCOMPLEX score is more favorable), respectively for SPRING, or 7.6 and 0.0, respectively for SPRINGNEG. The values of these weights were derived on the optimization of 5-fold cross validation performance against a gold standard interaction training

set (see “Pipeline training and benchmarking”). The score used in the final classifier is the SPRINGscore of the top-ranked dimer model.

In order to construct the database of template dimer structures, a list of all current Protein Data Bank (PDB)[38] entries was obtained using the database’s RESTful API. For each entry containing at least one protein chain, all “biomolecular assembly” structures were collected. From each biomolecular assembly, pairs of interacting proteins were extracted if both protein chains were at least 30 amino acids long (thus excluding protein-peptide interactions) and if there existed at least 10 interchain contacts (i.e., if there exist at least 10 interchain pairs of C $\alpha$  atoms whose distance is less than 8Å). Each pair is then split into monomeric chains and clustered at 70% sequence identity using CD-HIT.[39] Non-redundant protein pairs are derived from these clusters by pairing each cluster of monomers and extracting one pair at random that exists between the members of each cluster. Additional pairs are extracted if they are structurally non-redundant to the already extracted pairs, i.e., if the new pair has a TM-score < 0.5 by MM-align.[40] Clusters are also paired against themselves to allow for the extraction of homodimeric structures. This procedure resulted in a total of 48,591 non-redundant dimer template structures, constructed from 83,765 monomer chains. In order to make this structure database threadable, an HMM was constructed for each sequence by HHblits[35] using a February 2016 release of the UniProtKB database[41] clustered at 20% sequence identity. The resulting HMM was combined with secondary structure information as determined by DSSP and formatted to be threadable by HHsearch. In addition to the dimer structure library, a negative structure library was constructed from the biomolecular assembly structures, consisting of protein pairs which were found to be in the same complex, but not in physical contact (i.e., no inter-atom distance between C $\alpha$  atoms on differing chains were <8Å apart). These pairs were filtered for non-redundancy in a similar

manner to the dimer structure library; however, they were filtered to be non-redundant not only to each other, but also to the original dimer structure library. This way, the negative structure library only consists of pairs which were never shown to interact in any PDB structure. This library consists of 24,568 pairs constructed from 10,708 monomer chains.

In many solved protein complex structures, only the domains which form the interface are present in the structure due to size limitations of the structure solving method. In order to keep this phenomenon from hindering template identification, threading-based domain division of the query is implemented in PEPPI (**Figure 2.4**). This procedure collects all query-template alignments above a given Z-score threshold, 8.5, and counts how many of them consist of a large gapped region (>80 AA in length) on one or both termini of the query. If more than half of the total alignments contain a gapped region, the protein is split in two at the optimal domain boundary, and the two domains are analyzed recursively through the same procedure. Recursion stops when either a sequence is found to not have a sufficient proportion of gapped alignments (i.e., is single domain), or to have less than 10 alignments above the Z-score threshold (i.e., is a “hard” threading target). The optimal domain boundary is defined as the average of positions at which the gap region ends (or starts) if only N-terminal (or C-terminal) gaps were found. If both N- and C- terminal gapped alignments are found, each N-terminal boundary is paired with each C-terminal boundary, the midpoint between each pair is determined, and the optimal boundary is defined as the average of these midpoints. For a given pair of query sequences, each domain from one sequence is paired with each domain of the other sequence, and SPRING analysis is done on each domain-domain pair. The highest score derived from domain-domain pairing is kept as the score for the entire query pair.

### ***2.5.2 PPI prediction by sequence interology (“SEQ” module)***

The “SEQ” module of PEPPI implements a simple process to identify similar complexes through sequence similarity. The first step of this module is to split apart the two chains of the query and use BLAST[22] to search them through a database of single chain sequences taken from PPIs. The dimer sequence library was constructed from interactions taken from the PSICQUIC service.[28] This service was queried using the “MI:0407 (direct interaction)” term (from the molecular interaction ontology[42]) and filtered to retrieve a set of interactions which were annotated to be direct, physical interactions. First, interactions which were not strictly protein- protein interactions were filtered out, including protein-ligand, protein-nucleotide, and protein- complex interactions. Second, amino acid sequences were retrieved for each of the interactions, as they are only presented in PSICQUIC as a set of database identifiers, not as definitive amino acid sequences. Most of this was accomplished using Uniprot’s ID retrieval tool, but sequences were taken directly from the database itself when possible. Finally, the sequences of the dimer structure library were also added to the dimer sequence library. The final dimer sequence library consists of 515,058 interactions in total. It should be noted that this database is redundant, as the same interaction can be represented in multiple databases within PSICQUIC using distinct identifiers. The result of the BLAST search step is a set of single-chain hits sorted by sequence identity to the query. In order to translate this single chain similarity into dimeric similarity, dimers are retrieved from the sets of monomeric hits. The overall score for the dimer is the harmonic mean of the two respective monomeric sequence identities.

### ***2.5.3 PPI Prediction by non-similarity features (“CT”/“STRING”)***

In order to bolster the pipeline’s predictions in the absence of detected sequence or threading hits, PEPPI also uses a neural network model based on the conjoint triad (CT) feature.[14] This feature translates a pair of amino acid sequences into a vector based on the

frequency of amino acid triads in each of the two sequences. First, each amino acid of the input sequence is binned into one of 7 bins based on properties such as volume and dipoles, leading to a total of  $7*7*7=343$  possible different triplets. Then, a sliding window of length 3 is applied across the entire input sequence, counting each occurrence of amino acid triads, thus populating the conjoint triad vector for a single sequence. After normalizing the raw counts to constrain triad to be between zero and one, the two sequences' 343-entry vectors are concatenated into one 686-entry vector, which is then used as input in the neural network. The neural network is implemented as a Multi-Layer Perceptron Classifier in the scikit-learn package,[43] with a single hidden layer of size 1000. The model was trained using vectors generated from sequences of 27,967 interacting pairs from the dimer structure dataset detailed in the SPRING module, as well as an equal number of randomly paired chains. These 27,967 pairs were selected such that they were non-redundant to the test set (see 2.5.4). The final interaction probability output from this neural network is used in the consensus classifier.

While the primary focus of this pipeline is to predict direct, physical interactions, functional association data can also assist in classification. This functional association data comes from the interaction scores in the STRING v11 database.[23] In particular, the scores used are gene neighborhood (how proximal two genes are in the genome sequence), gene fusion/fission (if other organisms have orthologs of the interacting proteins which have fused into the same polypeptide), co-expression (correlation of gene expression levels across different cell states), and co-occurrence (occurrence of orthologs for each protein in other organisms, regardless of whether they're known to interact). To retrieve these four scores, the query proteins are monomerically searched through the sequences of the STRING database using BLAST, and any hits with >90% sequence identity are kept. The hits for each chain are then paired against

each other and searched through STRING links for a corresponding entry; the four corresponding scores for the first discovered database entry are passed to the classifier. If no hit is found, then missing values are passed to the classifier. It should be noted that the STRING database only contains information for intra-species interactions and does not provide any information on inter-species interactions, such as those in the SARS-CoV-2/Human dataset. PEPPI therefore operates without the STRING module for the SARS-Cov-2/Human application presented in this manuscript.

#### **2.5.4 Pipeline training and benchmarking**

In order to evaluate the interaction likelihood based on the calculated score  $x$  for a given module in **Figure 2.1**, the  $x$  value is compared against the scores attained on training set consisting of 800 non-redundant interactions from the IntAct[19] database (annotated as “direct interaction” and with an MI-score of at least 0.7), and 800 non-redundant non-interactions from the Negatome 2.0[20] database. From the scores of the interaction and non-interaction sets for the given module, two respective probability distributions were created by Gaussian kernel density estimation. From these reference distributions, the likelihood of interaction is expressed as the log-likelihood ratio:

$$\log(LR) = \log(p(x|I)) - \log(p(x|NI)) \quad (2)$$

where  $p(x|I)$  and  $p(x|NI)$  are the conditional probabilities at the score  $x$  in the interacting and non-interacting reference distributions, respectively. The overall likelihood for a given query pair is the sum of log-likelihoods for all separate modules.

The PEPPI pipeline was then tested on an independent test set consisting of 798 pairs from the dimer structure library, and 798 pairs from the negative structure library, both of which are non-redundant to the pairs in the training set (<50% sequence identity). In order to assess the

performance of the pipeline fairly, several non-redundancy measures have been put in place for the benchmark. For the SPRING and SEQ modules of PEPPI and for PRISM, any single-chain templates which have a sequence identity >50% to the query were removed from consideration. In order to ensure the test set was non-redundant to the training sets of competing programs and that this training set is appropriate for species-agnostic physical interaction prediction, we constructed a custom training set of 27,967 pairs from the dimer structure library such that they had <50% sequence identity to the test set pairs and an equal number of non-interacting pairs randomly pairing chains from this interaction set. This training set was subsequently used for training SPRINT, PIPR, D-SCRIPT, and the CT modules of PEPPI. Filtering criteria present in PRISM were disabled as the combination of these criteria and our own imposed template non-redundancy rules resulted in a high incidence of no templates being identified. 75 layers were used for PIPR instead of the default 50, as raising the layer number improved 5-fold cross validation performance on the provided training set. D-SCRIPT was trained according to program defaults.

#### ***2.5.5 SARS-CoV-2 virus and human host protein sequence collection***

The SARS-CoV-2 proteome was collected from the UniProtKB pre-release. Replicase polyprotein 1ab was split into nsp1-16 (excluding nsp11) according to its “chain” regions as described in the “Protein Processing” subsection of its UniProtKB entry; nsp11 was extracted from replicase polyprotein 1a in a similar fashion. As a result, the SARS-CoV-2 sequence set consisted of 31 protein sequences in total. The human proteome, consisting of 20,600 proteins, was also collected from the Uniprot database.[41] All  $20600 \times 31 = 638,600$  putative interactions were analyzed with PEPPI; any pairs resulting in a  $\log(\text{LR})$  greater than 0 were classified as interacting. The “gold standard” dataset is comprised interactions listed in the PSICQUIC

database[28] annotated as “direct interaction” as of April 2021 between the proteins of either SARS-CoV-1 or SARS-CoV-2 and human proteins, a total of 128 interactions.

## 2.6 Acknowledgements

We would like to thank Dr. Gilbert S. Omenn for helpful discussion. This work is supported in part by the National Institute of General Medical Sciences (GM136422, S10OD026825), the National Institute of Allergy and Infectious Diseases (AI134678), and the National Science Foundation (IIS1901191, DBI2030790, MTM2025426). The work in this manuscript was performed in part using the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by the National Science Foundation (ACI1548562).

## 2.7 Figures

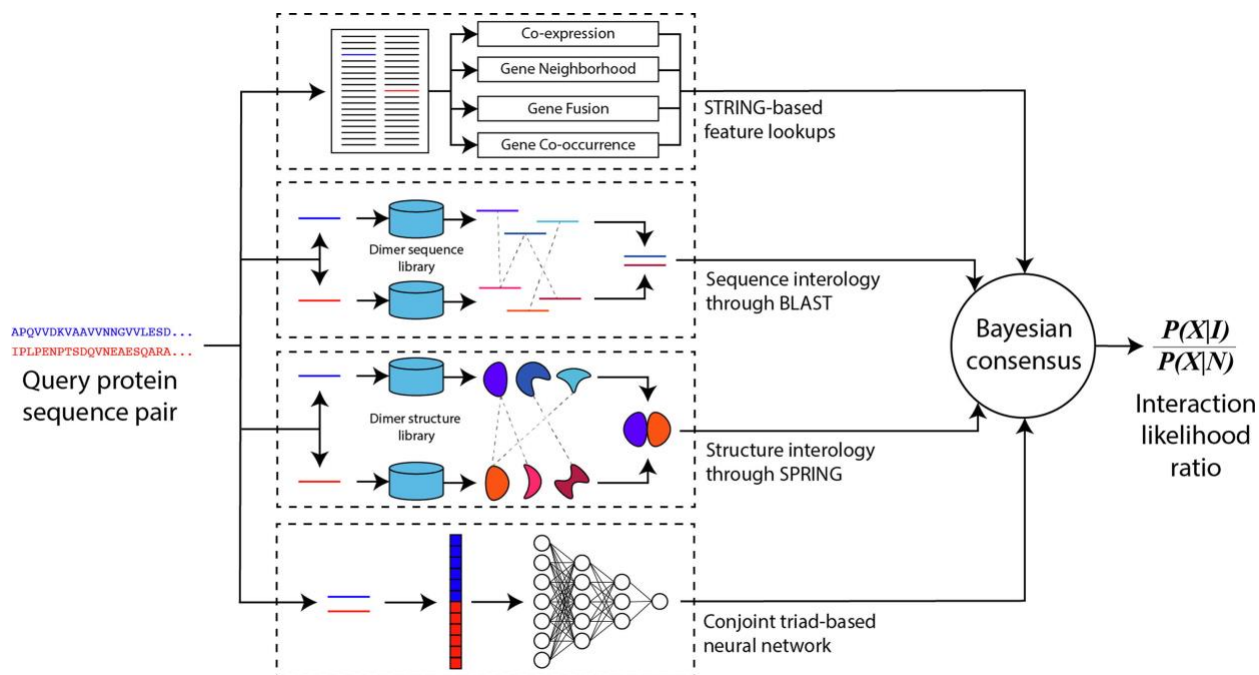


**Figure 2.1** An illustration of the PEPPi pipeline. This pipeline functions by analyzing a pair of input sequences via a series of independent modules, including structure similarity, sequence similarity, neural network classification, and functional association data. These modules are combined using a naïve Bayesian consensus classifier, which provides the final interaction score as a log-likelihood ratio.

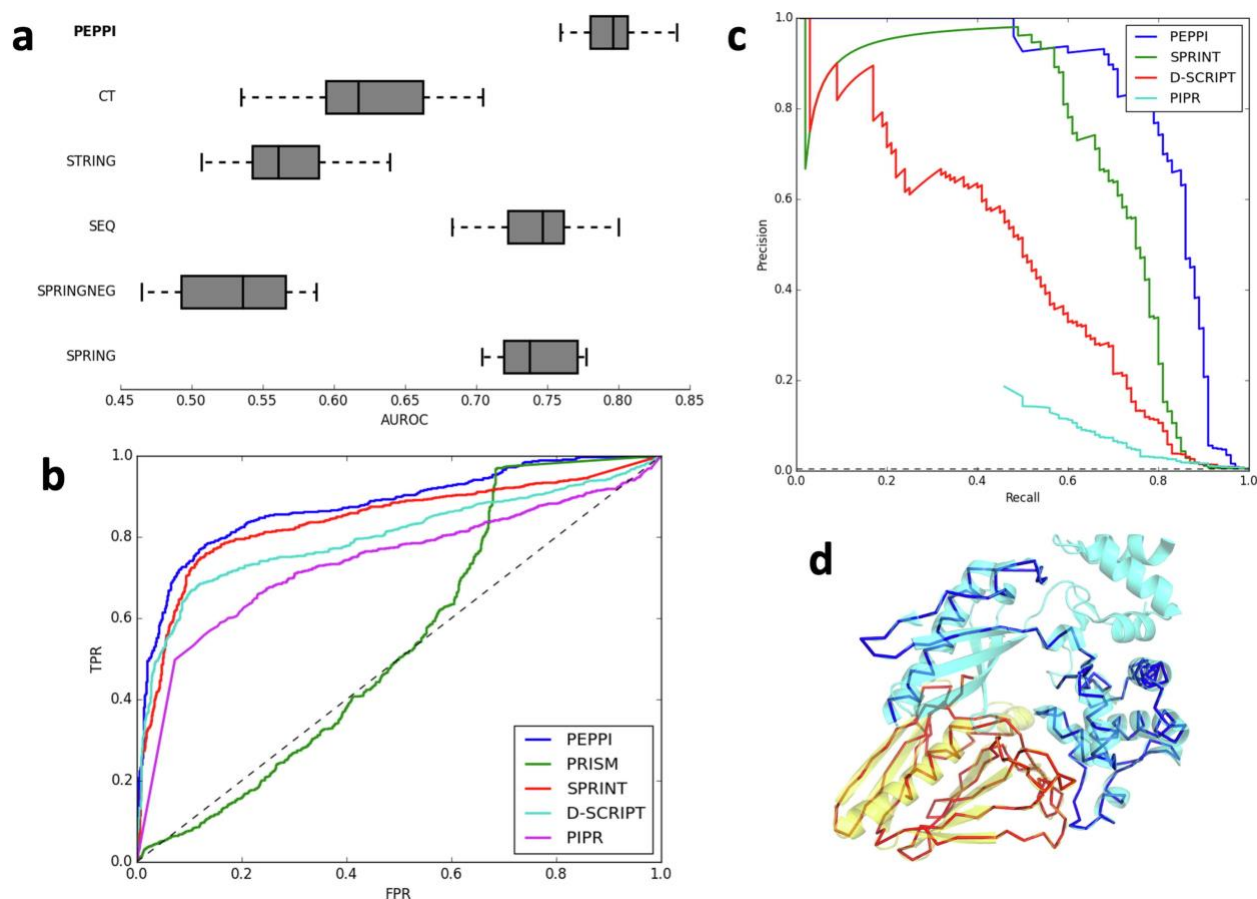


**Figure 2.2** PEPPi benchmark results. (a) 10-fold cross validation AUROC reveals that the full PEPPi pipeline outperforms its component modules: the neural network classifier (CT), the functional association data (STRING), the sequence similarity method (SEQ), the structure similarity method (SPRING), and the non-interaction similarity method (SPRINGNEG). (b) An ROC curve of the performance of PEPPi against PRISM, a structure similarity-based method, SPRINT, a sequence similarity-based method, PIPR, a deep learning-based method, and D-SCRIPT, a structure-aware deep learning-based method on a balanced testing set. The dotted line represents the performance of random classification. (c) Precision-recall curve of the performance of PEPPi against several other comparable programs on an unbalanced testing set. The dotted line represents the performance of random classification. (d) A superposition of an example dimer model (PDB 3CI0; chain J in red, chain K in blue) on its dimer template structure (PDB 5VTM; chain W in yellow, chain X in cyan).

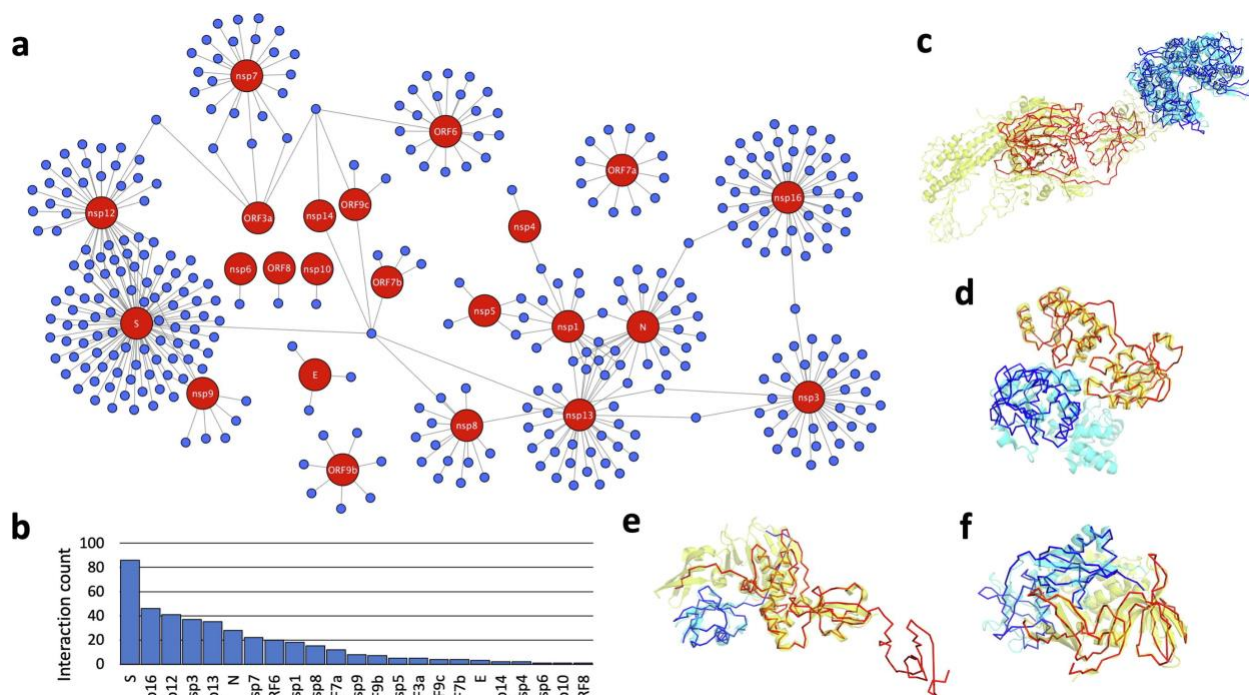


**Figure 2.3** A summary of SARS-CoV-2/human interactome prediction. (a) A network overview of the full interactome of human/SARS-CoV-2 protein pairs, with SARS-CoV-2 proteins colored red and human proteins colored blue. (b) A bar chart of the number of predicted interactions involving each SARS-CoV-2 protein. Proteins which were not predicted to have any interactions were excluded. (c) A superposition of a dimer model of the top-ranked SARS-CoV-2 Spike (in red) and Human ACE2 (in blue) interaction on its dimer template structure (PDB 6ACG; chain D in cyan, chain C in yellow). (d) A superposition of a dimer model of the SARS-CoV-2 nsp3 macrodomain (in red) and human PARP15 macrodomain (in blue) on its dimer template structure (PDB 2W2G; chain A in cyan, chain B in yellow). (e) A superposition of a dimer model of a domain of the SARS-CoV-2 nsp3 (in red) and human NEDD8 (in blue) on its dimer template structure (PDB 5WFI; chain C in cyan, chain A in yellow). (f) A superposition of a dimer model of a domain of the SARS-CoV-2 nsp3 (in red) and human UBD (in blue) on its dimer template structure (PDB 6BI8; chain C in cyan, chain A in yellow).

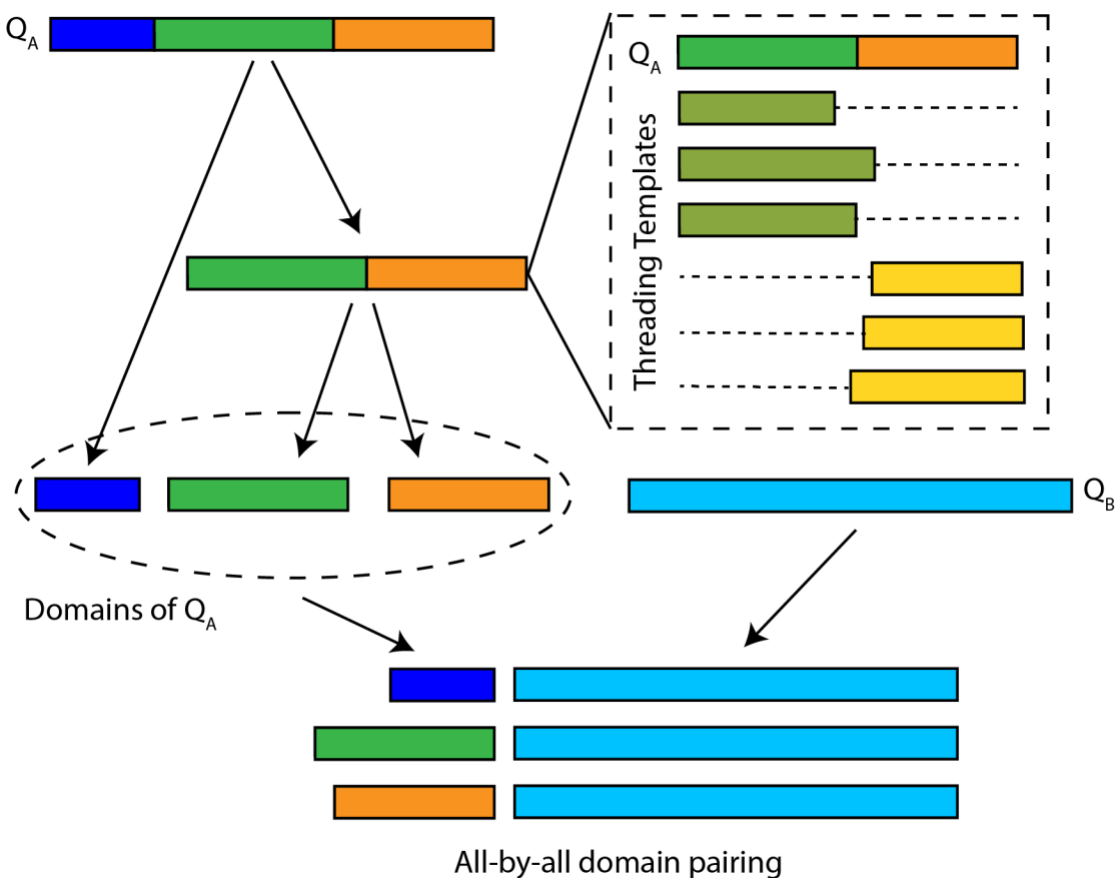


**Figure 2.4** An illustration of the domain division procedure for structure-based interology. In this figure, query sequence A ( $Q_A$ ) is divided into domains according to the domain boundary derived from the alignments of its threading templates. After domain division, all domains of sequence A are paired with the single-domain query sequence B ( $Q_B$ ) and analyzed individually.

## 2.8 Tables

|          | <b>AUROC</b>    | <b>Avg. Precision</b> | <b>Max MCC</b>  |
|----------|-----------------|-----------------------|-----------------|
| PEPPI    | <b>0.879</b>    | <b>0.895</b>          | <b>0.654</b>    |
| PRISM    | 0.544 (3.7e-80) | 0.511 (6.2e-115)      | 0.373 (6.9e-30) |
| SPRINT   | 0.841 (5.3e-7)  | 0.853 (6.0e-5)        | 0.632 (5.6e-2)  |
| D-SCRIPT | 0.800 (1.8e-15) | 0.839 (7.3e-8)        | 0.579 (8.9e-6)  |
| PIPR     | 0.738 (4.2e-30) | 0.760 (2.7e-24)       | 0.469 (5.3e-18) |

**Table 2.1** Performance of the control pipelines against PEPPI in the balanced benchmark. P-values (calculated by 1000-iteration bootstrapping of the distribution of differences in performance between methods) are presented in parentheses.

|          | <b>Avg. Precision</b> | <b>Max MCC</b>  |
|----------|-----------------------|-----------------|
| PEPPI    | <b>0.829</b>          | <b>0.799</b>    |
| SPRINT   | 0.696 (9.3e-4)        | 0.729 (1.8e-2)  |
| D-SCRIPT | 0.454 (3.6e-15)       | 0.512 (5.6e-11) |
| PIPR     | 0.122 (4.7e-95)       | 0.287 (2.5e-45) |

**Table 2.2** Performance of the control pipelines against PEPPI in the unbalanced benchmark. P-values (calculated by 1000-iteration bootstrapping of the performance difference distribution) are presented in parentheses.

## 2.9 References

1. Bell EW, Schwartz JH, Freddolino PL, Zhang Y. PEPPI: Whole-proteome Protein-protein Interaction Prediction through Structure and Sequence Similarity, Functional Association, and Machine Learning. *J Mol Biol.* 2022; 167530. doi:10.1016/j.jmb.2022.167530
2. Scott DE, Bayly AR, Abell C, Skidmore J. Small molecules, big targets: Drug discovery faces the protein-protein interaction challenge. *Nature Reviews Drug Discovery.* Nature Publishing Group; 2016. pp. 533–550. doi:10.1038/nrd.2016.29
3. Bryan CM, Rocklin GJ, Bick MJ, Ford A, Majri-Morrison S, Kroll A V., et al. Computational design of a synthetic PD-1 agonist. *PNAS.* 2021;118: e2102164118. doi:10.1073/PNAS.2102164118
4. Huang X, Pearce R, Zhang Y. De novo design of protein peptides to block association of the SARS-CoV-2 spike protein with human ACE2. *Aging (Albany NY).* 2020;12: 11263–11276. doi:10.18632/aging.103416
5. Snel B, Lehmann G, Bork P, Huynen MA. STRING: a web-server to retrieve and display the repeatedly occurring neighbourhood of a gene. *Nucleic Acids Res.* 2000;28: 3442–4. Available: <http://www.ncbi.nlm.nih.gov/pubmed/10982861>
6. Fields S. High-throughput two-hybrid analysis: The promise and the peril. *FEBS J.* 2005;272: 5391–5399. doi:10.1111/j.1742-4658.2005.04973.x
7. Li Y, Ilie L. SPRINT: Ultrafast protein-protein interaction prediction of the entire human interactome. *BMC Bioinformatics.* 2017;18: 1–11. doi:10.1186/s12859-017-1871-x
8. Pitre S, Dehne F, Chan A, Cheetham J, Duong A, Emili A, et al. PIPE: A protein-protein interaction prediction engine based on the re-occurring short polypeptide sequences between known interacting protein pairs. *BMC Bioinformatics.* 2006;7: 1–15. doi:10.1186/1471-2105-7-365
9. Baspinar A, Cukuroglu E, Nussinov R, Keskin O, Gursoy A. PRISM: A web server and repository for prediction of protein-protein interactions and modeling their 3D complexes. *Nucleic Acids Res.* 2014;42: W285–W289. doi:10.1093/nar/gku397
10. Hosur R, Peng J, Vinayagam A, Stelzl U, Xu J, Perrimon N, et al. A computational framework for boosting confidence in high-throughput protein-protein interaction datasets. *Genome Biol.* 2012. doi:10.1186/gb-2012-13-8-r76
11. Gong W, Guerler A, Zhang C, Warner E, Li C, Zhang Y. Integrating Multimeric Threading With High-throughput Experiments for Structural Interactome of Escherichia coli. *J Mol Biol.* 2021;433: 166944. doi:10.1016/j.jmb.2021.166944
12. Garzón JI, Deng L, Murray D, Shapira S, Petrey D, Honig B. A computational

- interactome and functional annotation for the human proteome. *Elife*. 2016;5: 1–27. doi:10.7554/eLife.18715
13. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Highly accurate protein structure prediction with AlphaFold. *Nature*. 2021;596: 583–589. doi:10.1038/s41586-021-03819-2
  14. Shen J, Zhang J, Luo X, Zhu W, Yu K, Chen K, et al. Predicting protein-protein interactions based only on sequences information. *Proc Natl Acad Sci U S A*. 2007;104: 4337–4341. doi:10.1073/pnas.0607879104
  15. Guo Y, Yu L, Wen Z, Li M. Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences. *Nucleic Acids Res*. 2008;36: 3025–3030. doi:10.1093/nar/gkn159
  16. Hashemifar S, Neyshabur B, Khan AA, Xu J. Predicting protein-protein interactions through sequence-based deep learning. *Bioinformatics*. 2018;34: i802–i810. doi:10.1093/bioinformatics/bty573
  17. Chen M, Ju CJT, Zhou G, Chen X, Zhang T, Chang KW, et al. Multifaceted protein-protein interaction prediction based on Siamese residual RCNN. *Bioinformatics*. 2019;35: i305–i314. doi:10.1093/bioinformatics/btz328
  18. Sledzieski S, Singh R, Cowen L, Berger B. D-SCRIPT translates genome to phenome with sequence-based, structure-aware, genome-scale predictions of protein-protein interactions. *Cell Syst*. 2021;12: 969–982.e6. doi:10.1016/j.cels.2021.08.010
  19. Orchard S, Ammari M, Aranda B, Breuza L, Briganti L, Broackes-Carter F, et al. The MIntAct project - IntAct as a common curation platform for 11 molecular interaction databases. *Nucleic Acids Res*. 2014;42: D358–D363. doi:10.1093/nar/gkt1115
  20. Blohm P, Frishman G, Smialowski P, Goebels F, Wachinger B, Ruepp A, et al. Negatome 2.0: A database of non-interacting proteins derived by literature mining, manual annotation and protein structure analysis. *Nucleic Acids Res*. 2014;42: D396–400. doi:10.1093/nar/gkt1079
  21. Guerler A, Govindarajoo B, Zhang Y. Mapping monomeric threading to protein-protein structure prediction. *J Chem Inf Model*. 2013;53: 717–725. doi:10.1021/ci300579r
  22. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol*. 1990;215: 403–410. doi:10.1016/S0022-2836(05)80360-2
  23. Szklarczyk D, Gable AL, Lyon D, Junge A, Wyder S, Huerta-Cepas J, et al. STRING v11: Protein-protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Res*. 2019;47: D607–D613. doi:10.1093/nar/gky1131
  24. Organization WH. WHO Coronavirus (COVID-19) Dashboard. WHO Coronavirus (COVID-19) Dashboard With Vaccination Data. In: Who [Internet]. 2021 [cited 10 Nov 2021] pp. 1–5. Available: <https://covid19.who.int/>
  25. Hoffmann M, Kleine-Weber H, Schroeder S, Krüger N, Herrler T, Erichsen S, et al. SARS-CoV-2 Cell Entry Depends on ACE2 and TMPRSS2 and Is Blocked by a Clinically Proven Protease Inhibitor. *Cell*. 2020;181: 271–280.e8. doi:10.1016/j.cell.2020.02.052
  26. Johnson BA, Xie X, Bailey AL, Kalveram B, Lokugamage KG, Muruato A, et al. Loss of furin cleavage site attenuates SARS-CoV-2 pathogenesis. *Nature*. 2021;591: 293–299. doi:10.1038/s41586-021-03237-4
  27. Grunewald ME, Chen Y, Kuny C, Maejima T, Lease R, Ferraris D, et al. The coronavirus

- macrodomain is required to prevent PARP-mediated inhibition of virus replication and enhancement of IFN expression. *PLoS Pathog.* 2019;15: e1007756. doi:10.1371/journal.ppat.1007756
28. Aranda B, Blankenburg H, Kerrien S, Brinkman FSL, Ceol A, Chautard E, et al. PSICQUIC and PSISCORE: Accessing and scoring molecular interactions. *Nature Methods.* Nature Publishing Group; 2011. pp. 528–529. doi:10.1038/nmeth.1637
  29. Gordon DE, Jang GM, Bouhaddou M, Xu J, Obernier K, White KM, et al. A SARS-CoV-2 protein interaction map reveals targets for drug repurposing. *Nature.* 2020;583: 459–468. doi:10.1038/s41586-020-2286-9
  30. Shin D, Mukherjee R, Grewe D, Bojkova D, Baek K, Bhattacharya A, et al. Papain-like protease regulates SARS-CoV-2 viral spread and innate immunity. *Nature.* 2020;587: 657–662. doi:10.1038/s41586-020-2601-5
  31. Han K, Zhang J. Roles of neddylation against viral infections. *Cellular and Molecular Immunology.* Nature Publishing Group; 2018. pp. 292–294. doi:10.1038/cmi.2017.100
  32. Kumar R, Mehta D, Mishra N, Nayak D, Sunil S. Role of host-mediated post-translational modifications (PTMS) in RNA virus pathogenesis. *International Journal of Molecular Sciences.* Multidisciplinary Digital Publishing Institute; 2021. pp. 1–26. doi:10.3390/ijms22010323
  33. Hipp MS, Kalveram B, Raasi S, Groettrup M, Schmidtke G. FAT10, a Ubiquitin-Independent Signal for Proteasomal Degradation. *Mol Cell Biol.* 2005;25: 3483–3491. doi:10.1128/mcb.25.9.3483-3491.2005
  34. Basler M, Buerger S, Groettrup M. The ubiquitin-like modifier FAT10 in antigen processing and antimicrobial defense. *Molecular Immunology.* Pergamon; 2015. pp. 129–132. doi:10.1016/j.molimm.2015.04.012
  35. Steinegger M, Meier M, Mirdita M, Vöhringer H, Haunsberger SJ, Söding J. HH-suite3 for fast remote homology detection and deep protein annotation. *BMC Bioinformatics.* 2019;20: 1–15. doi:10.1186/s12859-019-3019-7
  36. Zhang Y, Skolnick J. TM-align: A protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res.* 2005;33: 2302–2309. doi:10.1093/nar/gki524
  37. Liu S, Zhang C, Zhou H, Zhou Y. A physical reference state unifies the structure-derived potential of mean force for protein folding and binding. *Proteins.* 2004;56: 93–101. doi:10.1002/prot.20019
  38. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, et al. The Protein Data Bank. *Nucleic Acids Res.* 2000;28: 235–242. doi:10.1093/nar/28.1.235
  39. Fu L, Niu B, Zhu Z, Wu S, Li W. CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics.* 2012;28: 3150–3152. doi:10.1093/bioinformatics/bts565
  40. Mukherjee S, Zhang Y. MM-align: A quick algorithm for aligning multiple-chain protein complex structures using iterative dynamic programming. *Nucleic Acids Res.* 2009;37: e83–e83. doi:10.1093/nar/gkp318
  41. Bateman A, Martin MJ, Orchard S, Magrane M, Agivetova R, Ahmad S, et al. UniProt: The universal protein knowledgebase in 2021. *Nucleic Acids Res.* 2021;49: D480–D489. doi:10.1093/nar/gkaa1100
  42. Hermjakob H, Montecchi-Palazzi L, Bader G, Wojcik J, Salwinski L, Ceol A, et al. The HUPO PSI's Molecular Interaction format - A community standard for the representation of protein interaction data. *Nature Biotechnology.* Nature Publishing Group; 2004. pp. 177–183. doi:10.1038/nbt926

43. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. J Mach Learn Res. 2011;12: 2825–2830. Available: <http://jmlr.org/papers/v12/pedregosa11a.html>

## **Chapter 3 Probiotic-Host Interaction Network Modeling and Virtual Reality Visualization**

### **3.1 Abstract**

Probiotic bacteria, such as *E. coli* Nissle and *L. rhamnosus* GG, have many beneficial effects for human gut health. However, the mechanisms by which these bacteria colonize and symbiose with human epithelial cells is not understood on a protein level. Therefore, we have predicted protein-protein interaction networks between these two bacteria and the human proteome using PEPPI and subsequently extract biologically novel interactions and perform GO term analysis to identify the most prevalent functional interactions. These biological systems can give rise to highly complex networks whose underlying structure can be difficult to visualize in a flat two-dimensional environment. Three dimensional and virtual reality spaces can be useful for researchers who study such complicated systems due to their ability to more faithfully and immersively represent the complexity of biological networks. To this end, we have developed FALCON, a VR-compatible 3D network visualizer developed in the Unity game engine. FALCON supports features such as multiple network states, edge weighting, and visual customization, with development planned to be ongoing beyond its initial launch.

### **3.2 Introduction**

The colonization of hosts by diverse microbial communities can impact the physiology of host organisms in major ways, including symbiotic exchange of nutrients[1,2] and modulation of host immune response to pathogens.[3,4] At the heart of the host-microbe relationship lies sophisticated networks of protein[5–7] and metabolite[8,9] interactions, and therefore, to fully

understand the biological functionality of this relationship, we must first gain a comprehensive view of these interaction networks. Due to the size and complexity of inter-species protein-protein interaction (PPI) networks, experimental characterization of each individual interaction is infeasible. Therefore, computational methods can be used to construct these large-scale networks with higher efficiency than experimental methods. Previously, we have developed one such computational method, PEPPI,[10] which predicts proteome-wide interaction networks through a mixture of structure and sequence homology, functional association, and machine learning.

A particularly biologically interesting set of host-microbe systems are those involving the colonization of the human gut by probiotic strains of bacteria. One such probiotic strain is *E. coli* Nissle (EcN), which has been shown to interact with human hosts in a variety of ways, including induction of defensin expression,[11] enhancement of mucosal integrity,[12] and modulation of host immune responses.[3] Another bacterial strain of interest is *Lactobacillus rhamnosus* GG (LGG), which also is beneficial to the human gut through the modulation of host immune physiology.[13] These two strains are of particular importance because they both modulate how the human gut interacts with microbes in general and have been shown to improve gut health and resistance to infection by pathogens; therefore, elucidating how these strains form PPIs during colonization can provide insight to the colonization mechanisms of other beneficial bacteria.

Due to the complexity of protein-protein interaction networks, as well as many other biological systems such as protein-protein interactions, metabolomics,[14] etc., the visualization of networks is essential to large-scale inference in many fields of biology. Many of these fields rely on tools such as Cytoscape[15] or Graphviz[16] to visualize their networks, which are

restricted to flat 2D images that may potentially struggle to represent the full complexity of networks due to their low dimensionality. As a remedy to this issue, networks can be visualized in three dimensions. In particular, virtual reality (VR) poses an interesting possibility for network visualization because the immersive environment allows networks to be interrogated and manipulated as if they were real and tangible objects. While other tools exist in this regard, many focus on one discipline[17] or tend to offer more features than are needed for the average user.[18] To fill this gap, we have developed Free And Lightweight Computerization Of Networks (FALCON), which visualizes non-directional networks in 3D virtual reality space, as well as allowing more traditional mouse and keyboard controlled mode for users without VR helmets. Binaries for Windows 10, Mac OS X, and Linux operating systems as well as source code for FALCON can be found at <https://github.com/ewbell94/FALCON>. This program provides an important tool for clarifying the structure of complex biological systems and will serve an expanding audience of both researchers and educators as development of new features, such as support for directed networks, self interactions, and asymmetric network states continues.

### 3.3 Results

#### 3.3.1 *E. coli* Nissle interaction network

From the *E. coli* Nissle proteome, we selected 493 proteins which were most likely to interact with the human proteome due to predicted functional annotations and secretory signal peptide presence. This resulted in a total of 10,155,800 pairs to be analyzed by PEPPI,[10] 2,261 of which passed the classification threshold of  $\log(\text{LR}) > 0$  (see Methods for more details). The overall structure of this network is visualized in **Figure 3.1a**. A large portion of these interactions (685 interactions, 30.3%) were predicted to involve EcN soluble cytochrome b562

(CybC) due to the fact that this protein possesses a structural homolog of itself complexed to a GPCR. Due to the high structural homology of GPCRs, PEPPI had difficulty distinguishing between truly interacting proteins of this family and false positives. The interaction with the highest logLR in this set is with the human beta-2 adrenergic receptor (ADRB2), but other relevant interactions may exist. Another large portion of the network includes the maltodextrin-binding protein MalE (319 interactions, 14.1%). While some of these interactions may be valid, it is difficult to tell which are biologically relevant due to the common usage of maltose-binding protein as an affinity tag for protein purification. While these particular hubs are likely populated with false positives, this is not necessarily the case for all bacterial hubs. For example, Ecotin also occupies a large number of interactions (100 interactions, 4.4%) due to its function as a serine protease inhibitor. Finally, there seem to be some portions of the network where a core of human proteins is binding to a shell of EcN proteins, which contrasts with the rest of the network structure in which EcN proteins are typically serving as the hub. One such cluster is a set of three human transferrins (melanotransferrin, lactotransferrin, and serotransferrin) complexed with a set of bacterial iron/siderophore binding proteins; this cluster exists due to a structural template of serotransferrin and *N. meningitidis*'s transferrin binding protein A (PDB ID 3V89). The existence of this cluster is important because enhanced iron acquisition is a mechanism by which EcN exerts its beneficial effect of preventing pathogenic bacteria colonization.[19] Therefore, any potential modulation of that activity by the human host warrants further investigation.

The degree distribution for the top 30 most partnered EcN proteins is presented in **Figure 3.1a**. As is implied by the ambiguous names of some of these proteins, many of them remain poorly functionally understood; we can gain functional insight into these proteins based on the

functional annotations of their human binding partners. This approach is especially compatible with predicted networks since functional insight can be gained using protein pairs that have a propensity to interact whether they interact *in vivo*. To this end, we retrieved the GO term annotations of each human protein and scored the annotations based on the logLR of the interactions with these proteins. One interesting result from this analysis belongs to EcN heat-inducible protein HsIJ, whose function is completely unknown and whose combined (sequence & PPI) MetaGO[20] predictions only confidently (C-score > 0.9) label the protein with “membrane” (GO:0016020). However, all 21 predicted binding partners were annotated with “serine-type endopeptidase activity” (GO:0004252, C-score=1.0) and “serine hydrolase activity” (GO:0017171, C-score=1.0), and since the structural template used to find these interactions is an ecotin-trypsin complex (PDB ID: 1EZX) with HsIJ taking the place of Ecotin, it’s likely that HsIJ functions as a protease inhibitor. Another similar case is that of the uncharacterized DUF2207 domain-containing protein, whose function is unknown and MetaGO predictions only yield “membrane” (GO:0016020) and “plasma membrane” (GO:0005886). However, the binding partner functional annotations include terms such as “glycosylation” (GO:0070085, C-score=0.790), “glycosyltransferase activity” (GO:0016757, C-score=0.790), and “mannosylation” (GO:0097502, C-score=0.497), and since many of these interactions are based on structural homology to subunits of an oligosaccharyltransferase complex (PDB ID: 6S7O), it can be hypothesized that the DUF2207-containing protein has some glycosylation or glycan binding activity.

The most confident interaction of the predicted network (logLR=3.11, **Figure 3.1b**) is between human cytochrome c oxidase subunit 1 (COX1) and *E. coli* cytochrome bo(3) ubiquinol oxidase subunit 2 (CyoA) due to high structural homology (SPRINGscore = 34.24) and

reasonable sequence homology (sequence ID = 53.8%) to existing interactions. While this interaction is not likely to be biologically relevant, as COX1 is embedded inside the mitochondrial membrane and CyoA is embedded in the bacterial inner membrane, PEPPI indicates a high propensity for these proteins to interact, thus aligning with the pre-conceived evolutionary relationship between mitochondria and bacteria.[21] One particular interaction of interest is between EcN outer membrane usher protein HtrE and human CEP192 (logLR=0.041, **Figure 3.1c**). While CEP192 is localized within the cytosol of the human host cells,[22] it still is annotated with a “response to bacterium” GO term via orthology-based inference by Ensembl; HtrE, on the other hand, is localized in the bacterial outer membrane and is implicated in pilus assembly. Therefore, a predicted interaction between these proteins warrants further experimental investigation to understand whether it could be important to EcN-host adhesion. A similar interaction is predicted with EcN YqiG and CEP192, albeit with slightly lower logLR (0.040). Another interesting interaction of this dataset is the predicted interaction between uncharacterized lipoprotein YgeR and human PLAT3 (logLR=0.568, **Figure 3.1d**). PLAT3 is known to localize in the human cell membrane, and is used by picornaviruses for cell entry.[23] Given that PLAT3 is also annotated with “response to bacterium” and very little is known about YgeR except its localization on the bacterial outer membrane, this predicted interaction warrants further investigation to determine whether the interaction is biologically relevant and what purpose this interaction may serve. Finally, we predicted an interaction between predicted arylsulfatase AtsA and human ARSB, a protein specifically involved in the regulation of cell adhesion in the gut epithelium (logLR=0.994, **Figure 3.1e**).[24]

Finally, to gain a global view of the functional enrichment of interactions of the network as a whole, we annotated each interaction as the union of GO terms associated with the two

member proteins and determined the enrichment of each annotation in the set of true interactions using a hypergeometric distribution (**Figure 3.2**). The most enriched GO terms correspond to “iron ion binding” (GO:0005506, 27.1 fold enrichment,  $\log P = -1853.1$ ), “heme binding” (GO:0020037, 23.7 fold enrichment,  $\log P = -1711.0$ ) and “electron transfer activity” (GO:0009055, 17.8 fold enrichment,  $\log P = -1577.4$ ) due to the high number of interactions involving CybC. However, the enrichment with respect to iron binding is also partially due to the previously discussed transferrin cluster. Also among the most enriched terms, albeit with less statistical significance, is “arylsulfatase activity” (GO:0004065, 17.0 fold enrichment,  $\log P = -101.7$ ), a term associated solely with human proteins. With respect to depletion, the most depleted term is “cell aggregation” (GO:0098743, 0.03 fold enrichment,  $\log P = -29.9$ ), but there also is a noteworthy cluster containing “double-stranded DNA binding” (GO:0003690, 0.23 fold enrichment,  $\log P = -90.1$ ) and “DNA-binding transcription factor activity” (GO:0003700, 0.13 fold enrichment,  $\log P = -102.9$ ). Surprisingly, one of the most depleted terms is “biofilm formation” (GO:0042710, 0.04 fold enrichment,  $\log P = -27.7$ ), which is contrary to EcN’s known propensity to form strong biofilms.[25] This implies that the mechanism for EcN’s biofilm formation is facilitated through intraspecies interactions between EcN cells rather than through the interspecies interactions in this dataset.

### **3.3.2 *L. rhamnosus* interaction network**

From the *L. rhamnosus* GG (LGG) proteome, we selected 211 proteins which were most likely to interact with the human proteome due to predicted functional annotations and secretory signal peptide presence. This resulted in a total of 4,346,600 pairs to be analyzed by PEPPI,[10] 1,660 of which passed the classification threshold of  $\log(LR) > 0$ . The overall structure of this network is visualized in **Figure 3.3a**. Since LGG is not a model organism like *E. coli*, even less

is known about the functions of its proteome, and as a result, identification of the hubs is not very informative due to the lack of functional annotations associated with the hub proteins. However, 18 of the top 30 hubs (**Figure 3.3b**) are explicitly associated with ABC transporters, either being permeases, substrate-binding proteins, or the transporters themselves.

To better annotate the functions of the LGG proteins, we extracted GO terms of their binding partners using the same procedure as EcN. One result from this analysis involves the LGG DUF998 domain-containing protein (Uniprot ID: A0A180B5R4), which is only annotated to be membrane-localized. Its human binding partners are largely annotated with “transmembrane transporter activity” (GO:0022857, C-score=0.985), although the exact ligand of this transporter activity is unclear. The most likely candidate seems to be “amino acid transport” (GO:0006865, C-score=0.646), and given that this is annotated with the same level of confidence as several other amino acid transport carrier proteins of comparable interaction count (K8Q7Z7, C-score=0.650; A0A7S7JFL4, C-score=0.619), it is reasonable to assert that the DUF998-containing protein is involved in amino acid transport. Another interesting protein is the LGG lipoprotein (Uniprot ID: A0A249N8Q9), which is only annotated as being “membrane” localized (GO:0016020) and confidently predicted by MetaGO to have “transporter activity” (GO:0005215). Our analysis finds that the binding partners of this protein are located at a “tight junction” (GO:0070160, C-score=0.837) and is involved in “cell-cell adhesion via plasma-membrane adhesion molecules” (GO:0098742, C-score=0.837). Terms even as specific as “movement in host” (GO:0044000, C-score=0.364) appear in its binding partners. Therefore, this protein deserves investigation as a potential adhesion factor in gut colonization.

The top-ranked interaction of this network (logLR=1.93, **Figure 3.3c**) is between an LGG aminopeptidase (Uniprot ID: A0A807RP82) and human ERAP1 (Uniprot ID: Q9NZ08)

due to high structural homology (SPRINGscore = 42.88) to an existing interaction (PDB ID: 5LDS). Since ERAP1 is implicated in regulating the innate immune response,[26] this interaction could provide a mechanism by which LGG regulates the human immune response. An important interaction of this network is between an LGG cell surface protein (Uniprot ID: A0A807RPU5) and human TNFRSF12A (logLR=0.141, **Figure 3.3d**). Little is known about the LGG protein, but TNFRSF12A is known to promote cellular adhesion.[27] Therefore, this interaction may provide a mechanism for LGG adhesion to epithelial cells. Finally, while transferrins were an important part of the EcN network, only one interaction involving these proteins was detected in the LGG network: between a lipoprotein (Uniprot ID: A0A180A5E0) and serotransferrin (logLR=0.20, **Figure 3.3e**).

We analyzed the global functional enrichment of the whole network similarly to EcN (**Figure 3.2**). Many of the most enriched terms, such as “L-amino acid transmembrane transporter activity” (GO:0015179, 18.0 fold enrichment, logP = -244.9), “lipid transporter activity” (GO:0005319, 11.4 fold enrichment, logP = -279.5), and “cholesterol efflux” (GO:0033344, 20.8 fold enrichment, logP = -162.5), are due to the high number of interactions involving ABC transporters. However, not every type of transporter is enriched, as “carbohydrate import across plasma membrane” (GO:0098704, 0.15 fold enrichment, logP = -229.5), “fructose transmembrane transport” (GO:0015755, 0.03 fold enrichment, logP = -32.4), and “D-glucosamine PTS permease activity” (GO:0103111, 0.02 fold enrichment, logP = -56.0) are all depleted. Also contained within the enriched terms is “calcium-independent cell-cell adhesion via plasma membrane cell-adhesion molecules” (GO:0016338, 18.9 fold enrichment, logP = -124.8), “cytidine to uridine editing” (GO:0016554, 19.3 fold enrichment, logP = -102.1), and “glycerol kinase activity” (GO:0004370, 18.9 fold enrichment, logP = -82.8). The

enrichment of cell-cell adhesion proteins is especially striking and may reveal additional mechanisms important for LGG to colonize the human gut. Among the most depleted functions is “olfactory receptor activity” (GO:0004984, 0.03 fold enrichment,  $\log P = -35.8$ ), which draws a striking contrast to the EcN network in which this was one of the most statistically significant enrichments, likely due to the high number of GPCR-related interactions (7.8 fold enrichment,  $\log P = -491.3$ ). Terms involving kinases show a modulation of kinase-based signaling, as “kinase activity” (GO:0016301, 0.43 fold enrichment,  $\log P = -33.3$ ), in particular “protein serine/threonine kinase activity” (GO:0004674, 0.03 fold enrichment,  $\log P = -35.2$ ) is depleted overall, yet “receptor serine/threonine kinase binding” (GO:0033612, 8.9 fold enrichment,  $\log P = -49.9$ ) and “regulation of transmembrane receptor protein serine/threonine kinase signaling pathway” (GO:0090092, 4.1 fold enrichment,  $\log P = -68.5$ ) are both enriched. This enrichment is partly derived from a set of homodimer-templated interactions between human BMP10 and several LGG extracellular solute-binding proteins, implying that these LGG solute binding proteins could have potential to activate human signaling receptors.

### ***3.3.3 FALCON Visualization of probiotic networks***

In order to better interpret several structural features of the network, we visualized the predicted networks using the FALCON software. This visualization is presented in **Figure 3.4a** for EcN and **Figure 3.5a** for LGG, where human proteins have been colored blue, bacterial proteins have been colored red and labeled, and edges have been weighted according to their predictive confidence. It should be noted that in these figures, we are rendering the three dimensional view into a two dimensional projection, thus losing some of the clarity gained by modeling the network in three dimensions. As a point of comparison to traditional two dimensional visualization, we compare the visualization of several features of these networks

presented by Cytoscape’s “prefuse force directed OpenCL layout” to FALCON’s visualization. The first of these examples, shown in **Figure 3.4b**, shows a highly connected region of the EcN network involving both bacterial and host proteins. The traditional rendering of this portion of the network makes it difficult to distinguish connections between nodes, particularly because of the high density of connections. When visualized in 3D, this feature becomes a pseudo-cylindrical structure with alternating layers of EcN and human proteins, with the connections being formed between layers. There also exist a few branching features of this cylinder, such as the highly confident star-shaped structure connected through DegP. Through this visualization, the overlap and difference between the predicted binding partners of various outer membrane porin proteins is much clearer. In **Figure 3.4c**, we highlight the utility of the edge weighting feature of FALCON as we compare two connected clusters, one centered around Ecotin, and one centered around serine protease YdgD. The difference in weighting between these two clusters is striking, as the Ecotin cluster is much more highly confident than the YdgD cluster, which implies that the interactions involving Ecotin are much more likely to contain interesting biological insights as opposed to false positive red herrings. Also it should be noted that in the traditional rendering, these two clusters are spaced very far apart despite being connected, while these clusters are drawn together in the FALCON visualization.

For the LGG interaction network, the example highlighted in **Figure 3.5b** also has a layering structure of alternating bacterial and human proteins, but this aspect is visible in the traditional rendering. The clarity the FALCON visualization brings is the connections between the layers, which become spread out and visible through the formation of a six member ring in the middle human node layer. This helps give insight to the difference in connections and relative weights of the interactions involving the LGG PTS transporter proteins of this feature.

Finally, we present a particularly tangled example in **Figure 3.5c** in which the traditional representation of this portion of the network looks completely tangled and is generally uninformative as a result. While the FALCON visualization is still quite tangled, at the very least the node positions bring about a structure of two overlapping clusters and features such as the single human protein connected only to CCE29\_04775 and HMPREF0539\_2452 become visible.

### **3.4 Discussion and Conclusion**

The networks we have presented here provide novel biological insights into the protein interactions which occur during probiotic gut colonization, both in the form of annotating previously uncharacterized proteins and assigning novel behavior to more well-studied proteins. However, it should be noted that since this analysis is purely computational, the conclusions therein are hypotheses rather than confirmed biological phenomena. In addition, our survey of the predicted networks are far from exhaustive both inherently due to the sheer amount of information contained within the interaction set and because we have clearly missed some interactions known to occur (such as those involving toll-like receptors in immune modulation[28]). Therefore, this dataset is intended to be a source of discovery for other researchers doing small scale biochemical experiments rather than be the end of study regarding this topic. The PEPPI predictions will also be of great utility for cross-referencing with experimental XL-MS datasets currently being obtained on the same system.

FALCON provides useful analysis for those in the field of systems biology and other network-centric disciplines. Through the ability to easily visualize and compare network structures in an immersive environment, many features of these networks become apparent which were not as clear in traditional two-dimensional representations. FALCON also provides

an immersive view of biological networks that will be useful for scientific education and outreach. However, FALCON will not remain as it is described here, as development of the program is planned to be active; future planned features include support for directed graphs, visualization of self-interacting nodes, and allowing different nodes across network states. User feedback will be an essential part of this development to ensure that the features we add are valuable to the community as a whole. FALCON itself will remain lightweight in its implementation in order to remain accessible for outreach and educational contexts, but in order to provide more field-specific analytical power, we imagine an ecosystem of user-driven plugin capability which will allow the user to customize FALCON for their own specific purpose. For example, one could develop a PEPPI-specific plugin which provides detailed prediction information when hovering over an edge or visualizes both the structures of single protein nodes and the predicted structure of each interaction. Through the features users lay on top of the simple FALCON framework, FALCON will serve an ever-expanding audience of biological (and non-biological) scientists into the future.

### **3.5 Methods**

#### ***3.5.1 Sequence collection and PEPPI classification***

Amino acid sequences for the human, EcN, and LGG were taken from the Uniprot database.[29] For humans, UniProt makes a fasta file with one sequence for each of the 20,600 human genes easily available through the UniParc proteome UP000005640. For EcN and LGG, we began by collecting sequences from each strains' respective UniParc proteomes (4,801 sequences from UP000263938 for EcN, 2,877 sequences from UP000000955 for LGG). Since these UniParc sequences corresponded to multiple UniProtKB entries, each sequence was

labeled with the first corresponding UniProtKB entry, as these tended to be the most well-annotated for a given sequence.

In order to reduce the interaction search space, as well as focus the interaction modeling on proteins which were most likely to be biologically relevant, we filtered the bacterial proteomes with respect to two criteria: (1) Prediction of biologically relevant Gene Ontology (GO) terms. (2) Predicting the presence of signal peptides, which indicate a likelihood of secretion. For criterion 1, each sequence was analyzed using the sequence and PPI pipelines of MetaGO[20] (the structure pipeline was excluded to avoid the requirement of structurally modeling the entire bacterial proteomes), which assigned GO terms to each sequence with a confidence score (C-score) from the sequence pipeline alone, and the sequence pipeline combined with the PPI pipeline. A protein was kept using this method if it was annotated with a biologically relevant GO term with either C-score exceeding 0.9. These biologically relevant terms were GO:0043657 (host cell), GO:0019867 (outer membrane), GO:0005576 (extracellular region), GO:0009288 (bacterial-type flagellum), and GO:0042597 (periplasmic space) for EcN and GO:0043657 (host cell), GO:0005576 (extracellular region), GO:0009288 (bacterial-type flagellum), GO:0042597 (periplasmic space), GO:0009274 (peptidoglycan-based cell wall), GO:0009275 (gram-positive-bacterium-type cell wall), GO:0030063 (murein sacculus), and GO:0005886 (plasma membrane) for LGG. For criterion 2, we used SignalP 5.0 server[30] to predict signal peptide likelihood. If a sequence was predicted to have any kind of signal peptide with at least 0.9 probability, it was kept.

Using these two criteria, we isolated 493 and 211 proteins for EcN and LGG, respectively. When paired with the human proteins, this resulted in an interaction search space of 10,155,800 pairs for EcN and 4,346,600 pairs for LGG. To handle this immense number of

pairs, we predicted interactions using PEPPI,[10] a homology-based consensus predictor capable of whole-proteome interaction prediction. Each protein pair was put through the PEPPI pipeline and scored via  $\log(\text{LR})$ ; pairs with a  $\log(\text{LR})$  greater than zero were considered as interactions.

### **3.5.2 GO Term analysis**

To collect the functions of the EcN proteins' human binding partners, we start by retrieving the GO annotations contained within each sequence's UniProtKB entry using the QuickGO[31] REST API. We also annotate each sequence with the parent terms of each annotation. Each annotation is then scored using a C-score calculated by dividing the total amount of  $\log\text{LR}$  of predicted interactions whose binding partners possess a given GO term by the total amount of  $\log\text{LR}$  for all interactions involving that EcN protein.

To interrogate the functional enrichment of the predicted interactions, we re-use the retrieved annotations for each sequence described in the previous step. We then convert these annotations into a sparse binary matrix whose rows are all the analyzed protein pairs (both positive and negative) and columns are all the GO terms belonging to at least one sequence. An entry of this matrix takes a value of one if the corresponding GO term occurs in either of the paired proteins (i.e., the interaction GO term annotation set is the union of the individual protein annotation sets). In order to analyze the functional enrichment using this matrix, we modeled the problem using a hypergeometric distribution, where the "sample" is the set of protein pairs possessing a given GO term relative to the set of all pairs, and the "success" criterion is whether the given protein pair is positive (i.e.,  $\log(\text{LR}) > 0$ ). By modeling the problem in this way, we can then calculate the enrichment by dividing the observed number of positives by the expected mean of the hypergeometric distribution (with a pseudocount of 1 added to the numerator and denominator to handle zero-fold enrichment); p-values are calculated using the same distribution

(without pseudocounts) by summing the probability mass of the relevant tail (right tail for enhancement, left tail for depletion).

### 3.5.3 FALCON Implementation and Features

FALCON is implemented in the Unity engine,[32] a freely available game development engine with scripting in C#. Upon launching the software, it asks the user whether to launch in VR or in mouse and keyboard mode. The user can then load an undirected network using a simple interaction format (SIF) file or load the pre-installed example dataset discussed in the case study below. The SIF format, originally implemented in Cytoscape,[15] consists of a line-delimited list of interactions, each line consisting of three space-delimited columns: the first item of the interaction, a token describing the interaction (e.g., “pp” for “protein-protein”), and the second item of the interaction. When a network is loaded, the adjacency matrix is created from the SIF file, through which coordinates are calculated by Adam optimization[33] of the following function (Eqn 1):

$$f(c) = \sum_i \left[ D(c_i, \mathbf{c})^2 + \sum_{i \neq j} \left[ \frac{1}{D(c_i, c_j)^2} + A(i, j) * (D(c_i, c_j) - 2.0)^2 \right] \right] \quad (1)$$

where  $\mathbf{c}$  is the set of 3D coordinates of all nodes of the network,  $D(x,y)$  is the Euler distance between point  $x$  and  $y$ , and  $A$  is the adjacency matrix for the network. Conceptually, this function models connected nodes as harmonic springs with an ideal distance of 2 units (third term), ensuring equal spacing between all nodes by repulsing all nodes from each other (second term), and pulls all nodes towards the center of the network (first term). After the coordinates are calculated, they are scaled to remove overlapping node objects and translated so that the center is at the origin. Once coordinates have been determined, node objects are created at each coordinate point, node labels are assigned to each node, and edges are created to connect nodes

according to the adjacency matrix. Individual node positions can be altered after this step by grabbing and dragging the node to the desired location.

Loaded networks are saved through their adjacency matrix and node positions in memory as a network “state”. Additional network connectivities can be loaded using the options menu, but at present the new network state must contain the exact same nodes as the original network. Once loaded, the user can switch between loaded states at will. Also contained within the options menu is panels which allow for configuration of the nodes and edges of the network. The node configuration menu allows users to configure global node size, rename nodes using a comma separated values file (lines corresponding to oldName,newName), create node groups, and assign nodes to those groups. Nodes belonging to a group all share the same color and can have their node labels toggled on and off as one unit. The edge configuration menu allows the user to change the edge color, as well as re-weight the edges of the current network state using a comma separated values file (lines corresponding to nodeA-nodeB,weight).

### **3.6 Acknowledgements**

We would like to thank our collaborators Dr. Vincent Young, Dr. Philip Andrews, Dr. Manolo Plasencia, Dr. Rishu Dheer, and Catherine Barnier in their consultation regarding the results of this study. We would also like to thank the University of Michigan Museum of Natural History Science Communication Fellowship and the attendees of this fellowship’s “Scientist Spotlight” events, which helped us beta test FALCON and improve user experience. This work has been supported by the National Science Foundation (EF 2025426). The work in this chapter was performed in part using the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by the National Science Foundation (ACI1548562).

### 3.7 Figures

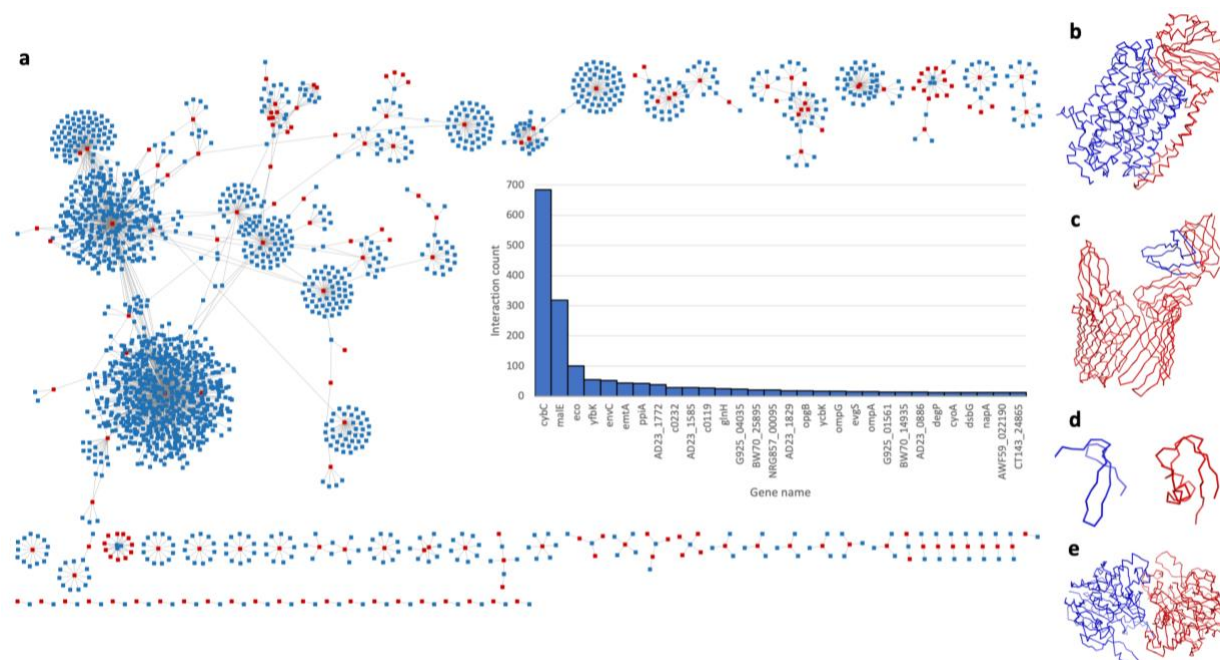


**Figure 3.1** Summary of the EcN interaction network. All blue color corresponds to human proteins, and all red color corresponds to bacterial proteins. Note that the shown structures are derived through threading and therefore structure quality is not necessarily indicative of interaction confidence. (a) The EcN network structure. Inset in this structure is the bacterial node degree distribution for the top 30 bacterial hubs. (b) PEPPI structure of CyoA-COX1, the top-ranked interaction. (c) PEPPI structure of HtrE-CEP192. (d) PEPPI structure of YgeR-PLAT3. (e) PEPPI structure of AtsA-ARSB.

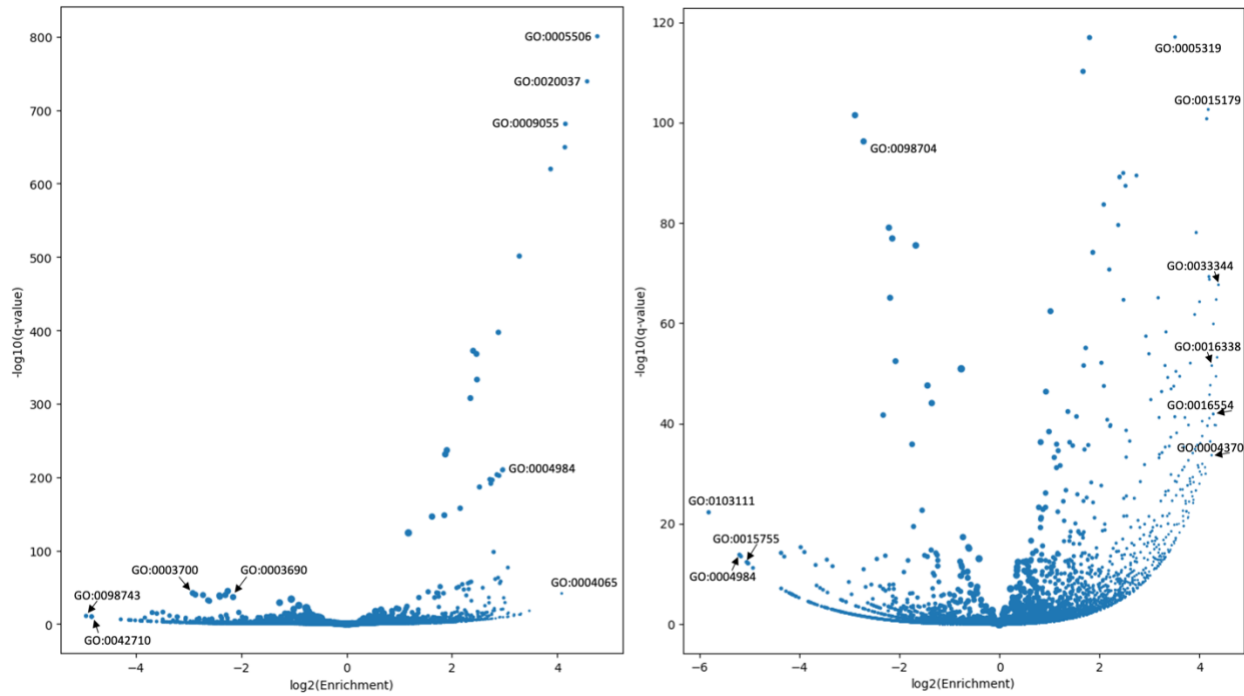


**Figure 3.2** Volcano plots of GO term enrichment for the EcN (left) and LGG (right). P-values are derived from a hypergeometric distribution; Q-values are calculated from these P-values by the Benjamini-Hochberg correction. Points are scaled in size corresponding to the number of protein pairs annotated with the given GO term.

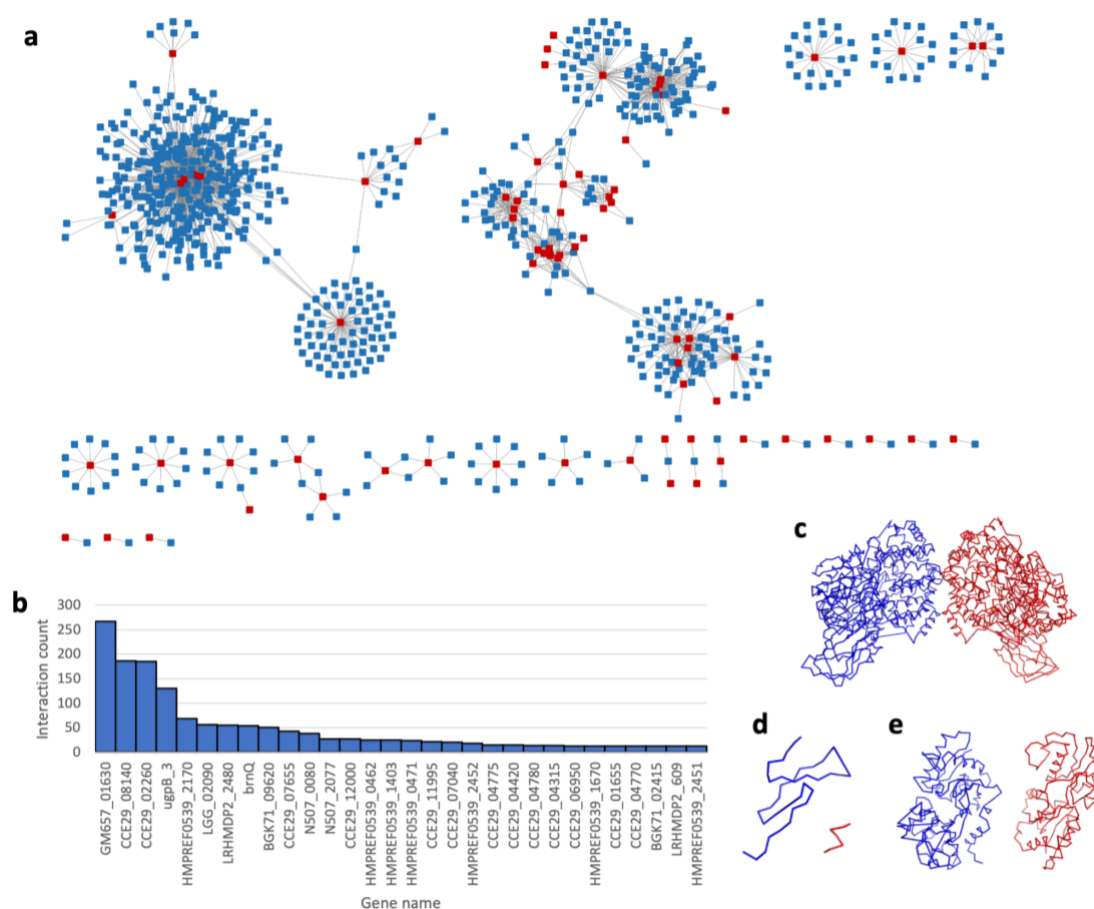


**Figure 3.3** Summary of the LGG interaction network. All blue color corresponds to human proteins, and all red color corresponds to bacterial proteins. Note that the shown structures are derived through threading and therefore structure quality is not necessarily indicative of interaction confidence. (a) The LGG network structure. (b) Bacterial node degree distribution for the top 30 bacterial hubs. (c) PEPPi structure of aminopeptidase-ERAP1, the top-ranked interaction. (d) PEPPi structure of cell surface protein-TNFRSF12A. (e) PEPPi structure of lipoprotein-serrotransferrin.

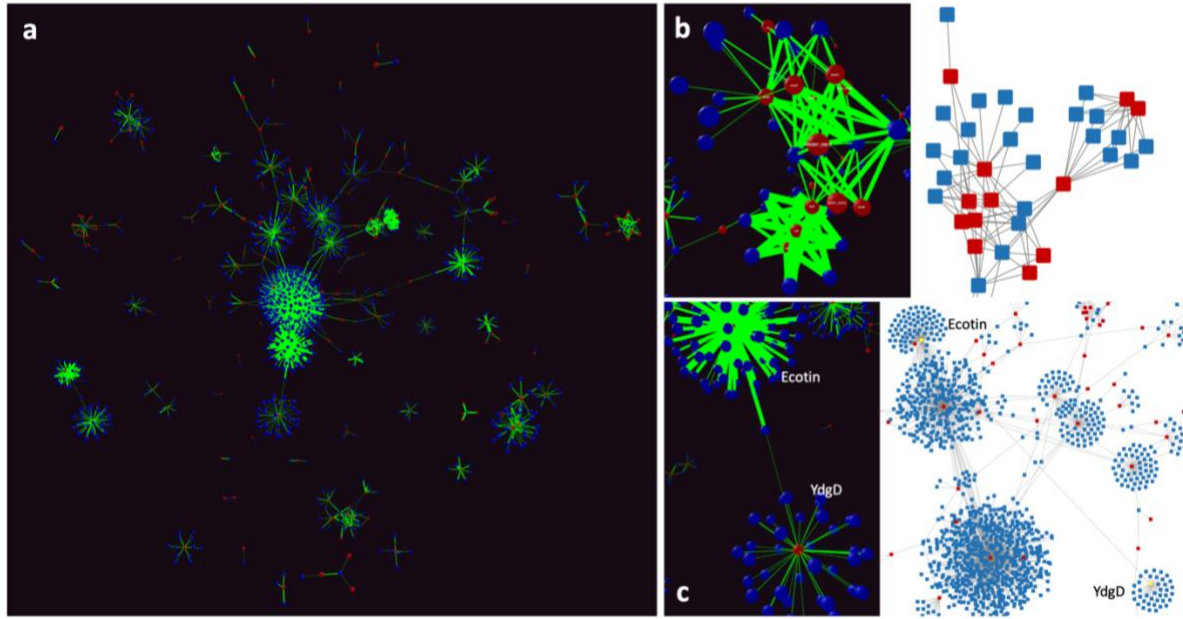


**Figure 3.4** FALCON visualization of the EcN interaction network. Blue nodes correspond to human proteins, red nodes correspond to bacterial proteins. (a) A view of the entire network visualized in 3D. (b) Pseudo-cylindrical feature visualized in FALCON side-by-side with its traditional two dimensional visualization. (c) Ecotin and YdgD clusters visualized in FALCON side-by-side with the traditional visualization of these clusters. In the traditional view, Ecotin and YdgD have been highlighted in yellow.

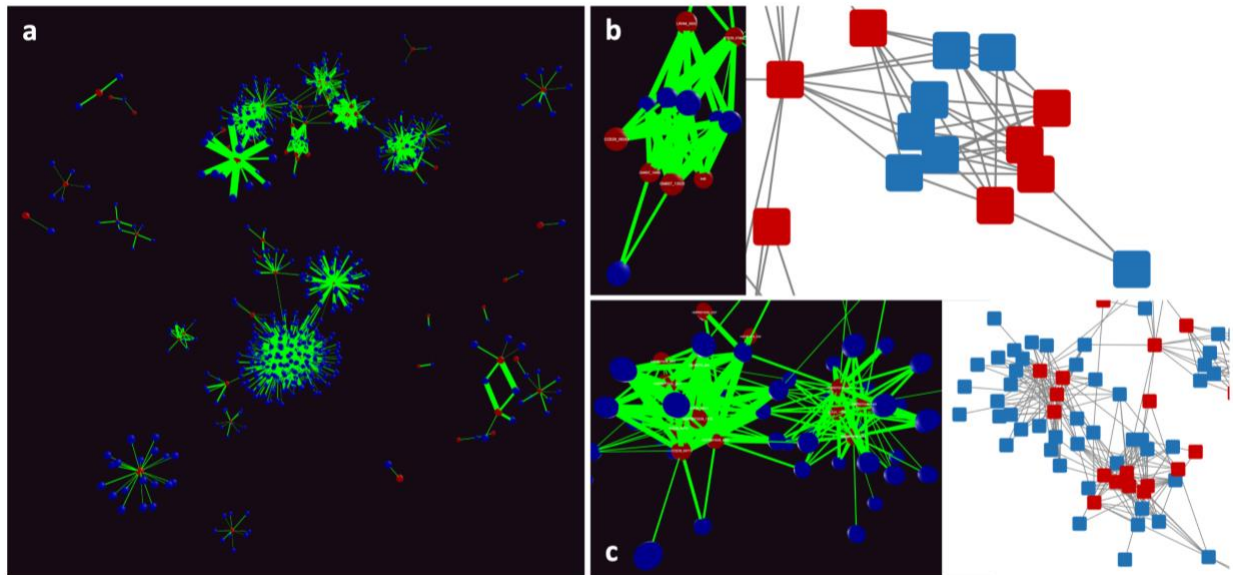


**Figure 3.5** FALCON visualization of the LGG interaction network. Blue nodes correspond to human proteins, red nodes correspond to bacterial proteins. (a) A view of the entire network visualized in 3D. (b) Pseudo-cylindrical feature visualized in FALCON next to its traditional two dimensional visualization. (c) Overlapping cluster pair feature visualized in FALCON next to its traditional two dimensional visualization.

### 3.8 References

1. Valentine AJ, Benedito VA, Kang Y. Legume Nitrogen Fixation and Soil Abiotic Stress: From Physiology to Genomics and Beyond. Annual Plant Reviews online. Chichester, UK: John Wiley & Sons, Ltd; 2018. pp. 207–248. doi:10.1002/9781119312994.apr0456
2. Conly JM, Stein K. The production of menaquinones (vitamin K2) by intestinal bacteria and their role in maintaining coagulation homeostasis. Progress in food & nutrition science. 1992. pp. 307–343. Available: <https://europepmc.org/article/med/1492156>
3. Arribas B, Rodríguez-Cabezas ME, Camuesco D, Comalada M, Bailón E, Utrilla P, et al. A probiotic strain of escherichia coli, nissle 1917, given orally exerts local and systemic anti-inflammatory effects in lipopolysaccharide-induced sepsis in mice. Br J Pharmacol. 2009;157: 1024–1033. doi:10.1111/j.1476-5381.2009.00270.x
4. Lazar V, Ditu LM, Pircalabioru GG, Gheorghe I, Curutiu C, Holban AM, et al. Aspects of gut microbiota and immune system interactions in infectious diseases, immunopathology, and cancer. Frontiers in Immunology. Frontiers Media S.A.; 2018. p. 1830. doi:10.3389/fimmu.2018.01830
5. Nicod C, Banaei-Esfahani A, Collins BC. Elucidation of host–pathogen protein–protein interactions to uncover mechanisms of host cell rewiring. Curr Opin Microbiol. 2017;39: 7–15. doi:10.1016/j.mib.2017.07.005
6. Guven-Maiorov E, Tsai CJ, Nussinov R. Structural host-microbiota interaction networks. PLoS Comput Biol. 2017;13: e1005579. doi:10.1371/journal.pcbi.1005579
7. Weimer BC, Chen P, Desai PT, Chen D, Shah J. Whole cell cross-linking to discover host-microbe protein cognate receptor/ligand pairs. Front Microbiol. 2018;9: 1585.

- doi:10.3389/fmicb.2018.01585
8. Maslowski KM. Metabolism at the centre of the host–microbe relationship. *Clin Exp Immunol*. 2019;197: 193–204. doi:10.1111/cei.13329
  9. Visconti A, Le Roy CI, Rosa F, Rossi N, Martin TC, Mohnhey RP, et al. Interplay between the human gut microbiome and host metabolism. *Nat Commun* 2019 101. 2019;10: 1–10. doi:10.1038/s41467-019-12476-z
  10. Bell EW, Schwartz JH, Freddolino PL, Zhang Y. PEPPI: Whole-proteome Protein-protein Interaction Prediction through Structure and Sequence Similarity, Functional Association, and Machine Learning. *J Mol Biol*. 2022; 167530. doi:10.1016/j.jmb.2022.167530
  11. Schlee M, Wehkamp J, Altenhoefer A, Oelschlaeger TA, Stange EF, Fellermann K. Induction of human  $\beta$ -defensin 2 by the probiotic *Escherichia coli* Nissle 1917 is mediated through flagellin. *Infect Immun*. 2007;75: 2399–2407. doi:10.1128/IAI.01563-06
  12. Ukena SN, Singh A, Dringenberg U, Engelhardt R, Seidler U, Hansen W, et al. Probiotic *Escherichia coli* Nissle 1917 inhibits leaky gut by enhancing mucosal integrity. Fox D, editor. *PLoS One*. 2007;2: e1308. doi:10.1371/journal.pone.0001308
  13. Fang H, Elina T, Heikki A, Seppo S. Modulation of humoral immune response through probiotic intake. *FEMS Immunol Med Microbiol*. 2000;29: 47–52. doi:10.1111/j.1574-695x.2000.tb01504.x
  14. Yang Y, Wurtele ES, Cruz-Neira C, Dickerson JA. Hierarchical visualization of metabolic networks using virtual reality. *Proceedings - VRCIA 2006: ACM International Conference on Virtual Reality Continuum and its Applications*. 2006. pp. 377–381. doi:10.1145/1128923.1128992
  15. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, et al. Cytoscape: A software Environment for integrated models of biomolecular interaction networks. *Genome Res*. 2003;13: 2498–2504. doi:10.1101/gr.1239303
  16. Gansner ER, North SC. An open graph visualization system and its applications to software engineering. *Softw Pract Exp*. 2000;30: 1203–1233. doi:10.1002/1097-024x(200009)30:11<1203::aid-spe338>3.3.co;2-e
  17. Legeth O, Rodhe J, Lang S, Dhapola P, Wallergård M, Soneji S. CellexVR: A virtual reality platform to visualize and analyze single-cell omics data. *iScience*. 2021;24: 103251. doi:10.1016/j.isci.2021.103251
  18. Pirch S, Müller F, Iofinova E, Pazmandi J, Hütter CVR, Chietini M, et al. The VRNetzer platform enables interactive network analysis in Virtual Reality. *Nat Commun*. 2021;12: 1–14. doi:10.1038/s41467-021-22570-w
  19. Deriu E, Liu JZ, Pezeshki M, Edwards RA, Ochoa RJ, Contreras H, et al. Probiotic bacteria reduce salmonella typhimurium intestinal colonization by competing for iron. *Cell Host Microbe*. 2013;14: 26–37. doi:10.1016/j.chom.2013.06.007
  20. Zhang C, Zheng W, Freddolino PL, Zhang Y. MetaGO: Predicting Gene Ontology of Non-homologous Proteins Through Low-Resolution Protein Structure Prediction and Protein–Protein Network Mapping. *J Mol Biol*. 2018;430: 2256–2265. doi:10.1016/j.jmb.2018.03.004
  21. Sagan L. On the origin of mitosing cells. *J Theor Biol*. 1967;14: 255–74. doi:10.1016/0022-5193(67)90079-3
  22. Park SY, Park JE, Kim TS, Kim JH, Kwak MJ, Ku B, et al. Molecular basis for unidirectional scaffold switching of human Plk4 in centriole biogenesis. *Nat Struct Mol Biol*. 2014;21: 696–703. doi:10.1038/nsmb.2846

23. Staring J, Von Castelmur E, Blomen VA, Van Den Hengel LG, Brockmann M, Baggen J, et al. PLA2G16 represents a switch between entry and clearance of Picornaviridae. *Nature*. 2017;541: 412–416. doi:10.1038/nature21032
24. Bhattacharyya S, Tobacman JK. Arylsulfatase B regulates colonic epithelial cell migration by effects on MMP9 expression and RhoA activation. *Clin Exp Metastasis*. 2009;26: 535–545. doi:10.1007/s10585-009-9253-z
25. Hancock V, Dahl M, Klemm P. Probiotic *Escherichia coli* strain Nissle 1917 outcompetes intestinal pathogens during biofilm formation. *J Med Microbiol*. 2010;59: 392–399. doi:10.1099/jmm.0.008672-0
26. Cui X, Rouhani FN, Hawari F, Levine SJ. An aminopeptidase, ARTS-1, is required for interleukin-6 receptor shedding. *J Biol Chem*. 2003;278: 28677–28685. doi:10.1074/jbc.M300456200
27. Wiley SR, Cassiano L, Lofton T, Davis-Smith T, Winkles JA, Lindner V, et al. A novel TNF receptor family member binds TWEAK and is implicated in angiogenesis. *Immunity*. 2001;15: 837–846. doi:10.1016/S1074-7613(01)00232-1
28. Grabig A, Paclik D, Guzy C, Dankof A, Baumgart DC, Erckenbrecht J, et al. *Escherichia coli* strain Nissle 1917 ameliorates experimental colitis via toll-like receptor 2- and toll-like receptor 4-dependent pathways. *Infect Immun*. 2006;74: 4075–4082. doi:10.1128/IAI.01449-05
29. Bateman A, Martin MJ, Orchard S, Magrane M, Agivetova R, Ahmad S, et al. UniProt: The universal protein knowledgebase in 2021. *Nucleic Acids Res*. 2021;49: D480–D489. doi:10.1093/nar/gkaa1100
30. Almagro Armenteros JJ, Tsirigos KD, Sønderby CK, Petersen TN, Winther O, Brunak S, et al. SignalP 5.0 improves signal peptide predictions using deep neural networks. *Nat Biotechnol*. 2019;37: 420–423. doi:10.1038/s41587-019-0036-z
31. Binns D, Dimmer E, Huntley R, Barrell D, O'Donovan C, Apweiler R. QuickGO: A web-based tool for Gene Ontology searching. *Bioinformatics*. 2009;25: 3045–3046. doi:10.1093/bioinformatics/btp536
32. Unity Technologies. Unity Engine. 2022. Available: <https://unity.com/>
33. Kingma DP, Ba JL. Adam: A method for stochastic optimization. 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings. International Conference on Learning Representations, ICLR; 2015. doi:10.48550/arxiv.1412.6980

## **Chapter 4 Evaluation of Multiple Sequence Alignment Pairing Schemes for Interchain Coevolutionary Structure Modeling**

### **4.1 Abstract**

Many modern deep learning-based protein structure prediction methods leverage the phenomenon of inter-residue coevolution to infer physical contact between residue pairs. However, coevolutionary relationships become difficult to perceive when multiple proteins are considered due to the requirement that members of the two chains' multiple sequence alignments (MSAs) be paired in some meaningful way. To this end, we have benchmarked several novel methods of pairing MSAs based on the performance of downstream inter-chain distance prediction. Through the preliminary version of this benchmark, we determined that AlphaFold2 clearly outperforms previous distance predictors at predicting the interchain region. We also determined that species-based pairing outperforms alternative methods, yet a machine learning-based method still performs better than random pairing.

### **4.2 Introduction**

The field of protein structure prediction has experienced a revolution at the hands of coevolution-based deep learning methods. These methods use sophisticated neural network architectures to infer relationships between pairs of amino acid residues through analysis of multiple sequence alignments (MSAs) consisting of sets of homologous sequences. While coevolution-based structure predictors originally assumed the form of convolutional neural networks used to predict inter-residue contacts,[1,2] more recent tools have used transformer-

inspired architectures capable of producing structures directly.[3,4] However, most structure predictors are primarily focused on the prediction of single chain structures, yet many proteins not functional or even structured unless they are in complex with other proteins. The difficulty in predicting these multi-chain structures arises from the fact that the construction of an MSA can only be performed on a single contiguous sequence and therefore, if interchain coevolutionary information is to be obtained, the MSAs of the individual chains must be paired in some way.

Many attempts have been made to address the problem of coevolution-based multi-chain structural modeling. Pozzati et al.[5] presented a “fold and dock” method based on the coevolution-based spatial restraint predictor trRosetta,[6] whose design was previously used only for monomeric structure prediction. A more sophisticated method is AlphaFold-Multimer,[7] a version of the original AlphaFold2 architecture optimized for complex modeling. In addition, AlphaFold2’s main competitor, RoseTTAfold[3] has also demonstrated their ability to perform complex modeling. However, the method by which these methods generate paired MSAs in their modeling workflows is dependent on the species annotations of the sequences composing the monomeric MSAs. The most common approach of this kind is a “phylogenetic” approach developed by Zhou et al,[8] where sequences across MSAs are paired if they belong to the same species of origin. Alternative approaches have also been used, such as the “reciprocal best hit” approach,[9] which pairs sequences if two proteins in separate proteomes are the best scoring sequence match in each other’s proteomes. These approaches limit paired MSA diversity within a given species, exclude sequences that do not have an annotated species of origin (such as those collected from metagenomic datasets), and make inter-species MSA pairing difficult or

impossible. Therefore, we present a few alternative methods for paired MSA generation and benchmark them against the traditional species-based approach.

## 4.3 Results

### 4.3.1 Comparison of interchain distance predictors

Before we can begin establishing which MSA pairing method is superior, we must first choose which program we will use to predict interchain distance. Since there are no algorithms built explicitly for interchain distance prediction, we choose to predict the interchain region using single chain distance predictors such as DeepPotential,[10] trRosetta,[6] and AlphaFold2.[4] We chose to benchmark these programs on 1,000 pairs of heterodimeric structures from the PEPPI[11] structure dimer database. These programs predict a C $\beta$  distance map from a species-paired multiple sequence alignment concatenated as if the two chains were one continuous chain (with a 100-residue glycine spacer for AlphaFold2 only); the interchain distance map is isolated for evaluation. These programs were then benchmarked by the mean absolute error (MAE) of the mode of each residue pair distribution relative to the native interchain C $\beta$  distances. The residue pairs were ranked by total probability density under each program's given upper distance threshold (<20Å).

Since 33 of the heterodimer structures did not have coordinates for C $\beta$  atoms and several more were unable to finish prediction due to memory/time constraints, the results presented here are preliminary and based on the subset of interactions which completed (949 for DeepPotential, 960 for trRosetta, and 965 for AlphaFold2). For all comparisons (top-L/5, top-L/2, and top-L total distances where L is the sum of the two chain lengths), AlphaFold2 attained lower MAE with tremendous statistical significance (**Figure 4.1**). In addition, this difference is numerically

large, with a 28.0% and 26.9% lower average top-L MAE than DeepPotential and trRosetta, respectively (**Table 4.1**). Despite there being statistical significance between DeepPotential and trRosetta for the top-L distances, the actual difference in performance is only 1.5%. Therefore, DeepPotential and trRosetta practically offer the same performance, both of which are inferior to AlphaFold2.

While this benchmark analyzes the performance of these programs from a perspective of precision (i.e., the program is evaluated based on the accuracy of the residue pairs which the predictors deem most likely to be relevant distances), we also chose to benchmark these programs using a recall-inspired analysis (i.e., the program is evaluated based on the accuracy of distance predictions for residue pairs which are natively relevant distances). In this benchmark, referred to as the ground truth-based benchmark, we calculate MAE based on all residue pairs whose ground truth distance is less than the upper distance threshold ( $<20\text{\AA}$ ). It should be noted that an additional protein pair which returned no result for the original benchmark due to the lack of most (but not all) C $\beta$  atoms of the native structure has returned a result in this benchmark, and thus each ground truth-based plot and average MAE contains this additional data point. While the differences between these programs appear to be statistically significant, the average MAEs only vary by 0.01. A major contributor to this lack of difference in performance is the fact that the performance for a majority of the protein pairs does not change between the different programs, indicating that the predictors have predicted only the maximum allowable distance ( $20\text{\AA}$ ) for every interacting residue pair for these cases. Essentially, this means that the distance predictors did not predict an interface being formed using the ground truth interfacial residues. Despite these benchmark results, we proceed with benchmarking paired MSA methods using AlphaFold2.

### 4.3.2 Paired MSA generation benchmarking

In order to determine which MSA pairing method yields the most meaningful coevolutionary information, we evaluate the quality of interchain C $\beta$  distance maps predicted by AlphaFold2 using MSAs generated by each pairing method. The pairing methods being compared are SPEC, a traditional pairing of sequences based on the species of origin; CT, a shallow neural network model for classifying PPIs taken from PEPPI[11] and OPT, an algorithmic procedure which chooses pairs such that interchain coevolution is optimized (see Methods for the implementation details of each of these methods). We begin by first benchmarking these methods against a naïve baseline method, RAND, in which sequences are randomly paired together. The calculation of MAE in the pairing method comparison benchmarks differs from the previous benchmark in that the upper distance threshold is changed from 20Å to 22Å to match the AlphaFold2 distance binning scheme. In addition to the previously noted issues regarding a lack of C $\beta$  atoms in the native structure and resource restraints for distance predictions, even more structures did not produce a result due to unfinished paired MSA generation (particularly for OPT, which requires a great deal of covariance calculations that make the algorithm inefficient overall). Therefore, these results are quite preliminary and only represent the subset which completed predictions (965 for SPEC, 808 for CT, 159 for OPT, and 784 for RAND). When compared against the random baseline (**Figure 4.2, Table 4.2**), SPEC easily achieves statistically significantly higher performance, with a 11.2% higher average top-L MAE. CT does not attain higher performance than randomness for the traditional top L-based benchmarks but attains significantly higher performance according to the ground truth-based benchmark. Finally, OPT does not attain significantly higher performance than randomness for either the traditional or ground truth-based benchmarks.

After comparing each pairing method to the random pairing baseline, we also compare each method to each other (**Figure 4.3, Table 4.2**). Given the results of the comparison to randomness, one could hypothesize that the ordering of method performance would be SPEC (clearly superior to randomness), CT (superior to randomness in one of the two benchmarks), then OPT (not superior to randomness). This ordering does hold: SPEC is significantly better than CT, and CT is significantly better than OPT (which runs contrary to their overall average MAEs due to the wide difference in dataset coverage between the two). Strangely, SPEC is not significantly better than OPT except for the ground truth-based benchmark; this is likely an artifact of the relatively small number of points for OPT.

#### **4.4 Discussion and Conclusion**

As previously mentioned, the results presented in this chapter are preliminary and deserve a more complete benchmark before definite conclusions can be drawn. Based on the current results, it is relatively unsurprising that AlphaFold2 attains higher performance at predicting interchain C $\beta$  distances due to AlphaFold2's superiority in single chain modeling relative to previously developed methods.[4] However, despite its relative superiority, it does not predict interactions between interfacial residues in a majority of our tested cases. Therefore we should explore alternative predictors for this benchmark, such as RoseTTAfold[3] or the multiple chain-optimized AlphaFold Multimer.[7] Our benchmark indicates that species-based pairing seems to capture coevolutionary signals more effectively than the other tested methods, but this does not mean that the other methods are useless and that one should only use species-based pairing for this task. For example, one possibility for improving upon species-based pairing is based in the observation that CT performs better than random pairing and thus captures some interchain coevolutionary signal, but that signal is relatively weak. Therefore, a consensus

of the pairing methods explored here may provide a stronger signal and therefore higher-performing predictions. Another possibility is that the modes of pairing we have presented are too based in the assumption that we know what information will be important to the deep learning architecture of AlphaFold2 (i.e., what if AlphaFold2 isn't capturing coevolution in the way we think about it?). The solution to this problem would be to utilize a deeper neural network architecture than what was offered by the CT method, such as an architecture built on cross attention layers. This pairing module could be constructed independently but would likely be more effective if trained as part of the AlphaFold2 architecture in end-to-end training.

The methods discussed here have primarily focused on the field of multimer structure prediction, but these ideas could very simply be extended to the field of protein-protein interaction prediction. In this chapter, we have benchmarked MSA pairing on its ability to extract coevolutionary relationships when it is assumed to be present, but a reasonably effective pairing method could also be utilized to detect when coevolution is not present. As a result, the strength of coevolutionary signal between chains could be utilized to predict whether an interaction occurs between two chains, and if the coevolution detection is sufficiently sensitive, the affinity of the interaction could be predicted. However, before any of these pipelines are possible, we must first solidify what factors will lead to optimal paired MSA construction and thus high-fidelity protein complex structure modeling.

## **4.5 Methods**

### ***4.5.1 Dataset collection and benchmark***

In order to benchmark the paired MSA methods, we extracted 1,000 heterodimeric PPI PDB structures from the dimer template library from PEPPI.[11] MSAs were constructed out of

each single chain of these pairs using DeepMSA.[12] Monomeric MSAs were paired, with corresponding sequences concatenated directly (for DeepPotential and trRosetta) or spaced by a 100-glycine linker (for AlphaFold2 only). Insertion sequences were also preserved for AlphaFold2 predictions as it can utilize these features in prediction. These paired MSAs were then subsequently used to predict a complex structure using a modified version of AlphaFold2[4] which excludes the use of templates in structure prediction (thus requiring all information being supplied from the MSA), or a distogram from DeepPotential[10] or trRosetta.[6]

From the AlphaFold2 structure prediction, we extract the predicted C $\beta$  distogram to compare against the native protein structure (or use the DeepPotential/trRosetta predicted distogram). To assess the accuracy of a prediction relative to its native structure, we first order all interchain residue pairs by their total probability density below the 22Å cutoff (20Å for the distance prediction method benchmark for consistency with trRosetta and DeepPotential) and taking the top residue pairs scaled by the sum of chain lengths. From these top residue pairs, we then calculate the mean absolute error (MAE) between the mode of the distance histogram and the native interchain distance. We also perform a set of “ground truth-based” benchmarks wherein all inter-residue pairs with a ground truth distance beneath the upper distance threshold are considered in the MAE rather than ordering by total probability density.

#### **4.5.2 MSA pairing methods**

*Species-based pairing (SPEC).* The species-based pairing method, described previously by Zhou et. al.,[8] uses the species to pair sequences. Since the MSA is constructed by DeepMSA,[12] the sequences acquired in the Uniclust and Uniref databases are annotated with the species of origin; the sequences acquired from the metaclust stage are not used in pairing due to the lack of a clear species. Sequences are paired together if their species annotation is the

same; if more than one possible pair could be made, the sequences are sorted by sequence identity to the query and paired by rank until no more pairs can be formed.

*Conjoint Triad-based neural network pairing (CT).* The CT-based pairing method leverages the conjoint triad-based neural network module from PEPPI.[11] This module transforms amino acid sequences into a fixed length vector via the conjoint triad method[13] and classifies the vector as an interaction using a shallow neural network. To pair the MSA using this method, the monomeric MSAs are first filtered using hhfilter[14] at 70% sequence identity to promote MSA diversity, and sequences from metaclust are removed. The full sequences of each remaining MSA member are then retrieved from their respective source databases. These sequences are then paired in an all-by-all fashion and classified individually by the CT module. The pairs are ordered by their interaction probability, and a maximum of 10,000 interactions (in addition to the query sequence pair) above a probability threshold of 0.5 are kept as the final paired MSA.

*Coevolution maximization-based pairing (OPT).* The OPT-based pairing method is a novel coevolution optimization method which iteratively refines the paired MSA until the pairs optimize the total amount of interchain coevolution. This process begins by adding the query sequence pair to the paired MSA and constructing a set of candidate MSA pairs from an all-by-all pairing of the MSA member sequences filtered by hhfilter to 70% sequence identity (including metaclust sequences). These candidate pairs are then added to the paired MSA at random and kept in the paired MSA if the total covariance of the interchain region increases:

$$\left\{ \begin{array}{l} Cov = \sum_{i=1}^{L_1} \sum_{j=L_1+1}^{L_1+L_2} \sum_{a=1}^{22} \sum_{b=1}^{22} f(i, j, a, b) - f(i, a)f(j, b) \\ f(i, a) = \frac{1}{N + 22} \left( 1 + \sum_{n=1}^N \delta_{a, a_n^i} \right) \\ f(i, j, a, b) = \frac{1}{N + 22} \left( \frac{1}{22} + \sum_{n=1}^N \delta_{a, a_n^i} \delta_{b, b_n^j} \right) \end{array} \right. \quad (1)$$

After all candidate pairs have been initially evaluated, all candidate pairs are then challenged by removing pairs in the MSA and adding pairs not in the MSA and keeping all changes that optimize interchain coevolution. This process of challenging candidates iterates until the total coevolution changes by less than 1% or until a maximum number of challenge iterations has been reached:

$$I = floor \left( \frac{\sum_{m=N_A, n=N_B}^{min(m,n)=0} [m * n]}{N_A * N_B} \right) \quad (2)$$

*Random pairing (RAND).* The random pairing method begins by collecting the two chains' monomeric MSAs filtered by hhfilter, and pairing the query sequences. The remaining sequences are shuffled randomly and paired one-to-one until either MSA reaches its maximum depth. The resulting paired MSA is as deep as the shallower of the two monomer MSAs.

#### 4.6 Acknowledgements

This work was funded by the National Institute of Allergy and Infectious Diseases (R01AI13467801).

## 4.7 Figures

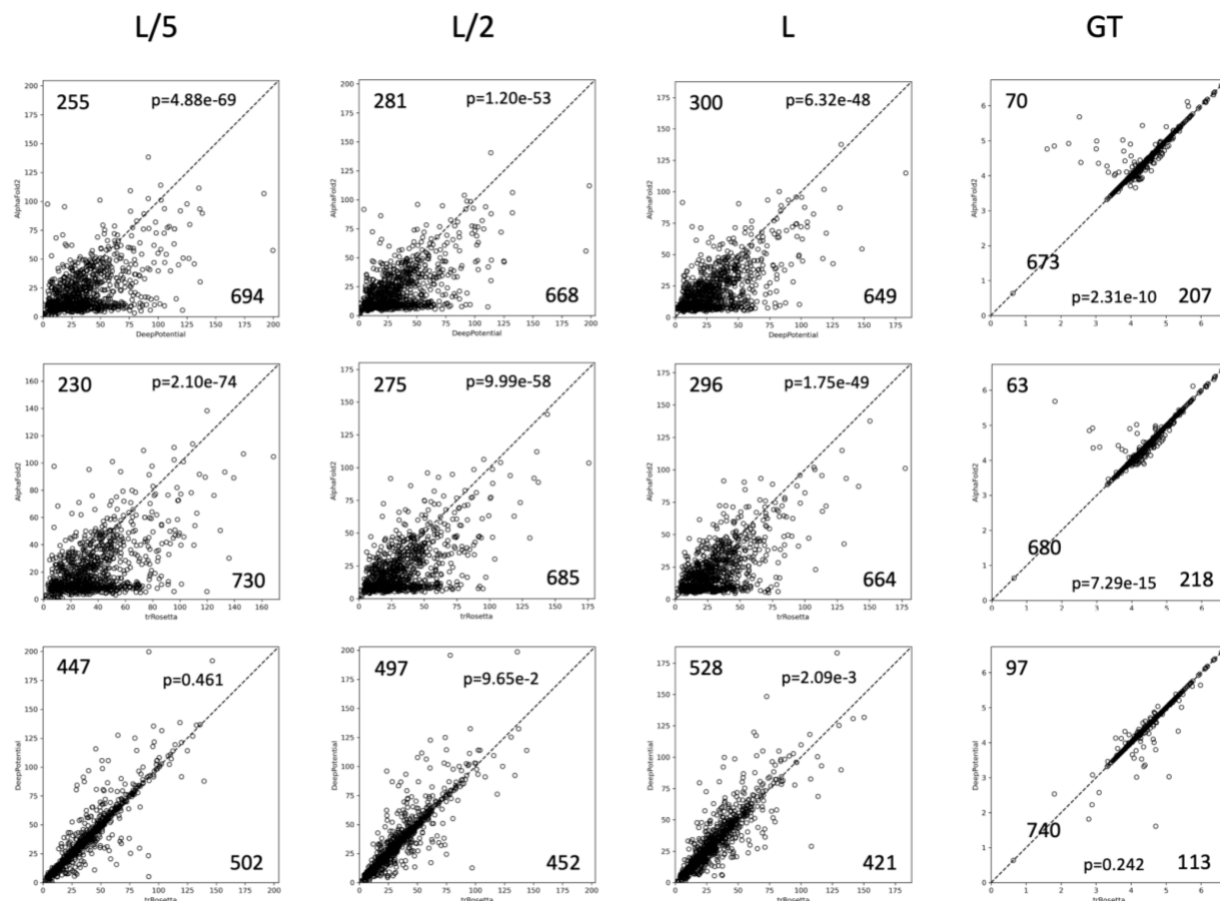


**Figure 4.1** Comparisons of the interchain distance prediction mean absolute error (MAE) for each of the tested benchmark programs. Rows correspond to each pairing between the three programs, columns correspond to different amounts of residue pairs considered in each point's MAE (where L is the sum of the two chain's lengths); the ground truth-based MAE is labeled as "GT". Each plot is marked with the total number of points above and below (and on for the GT column) the dotted line and a p-value calculated by a two-sided Wilcoxon signed-rank test.

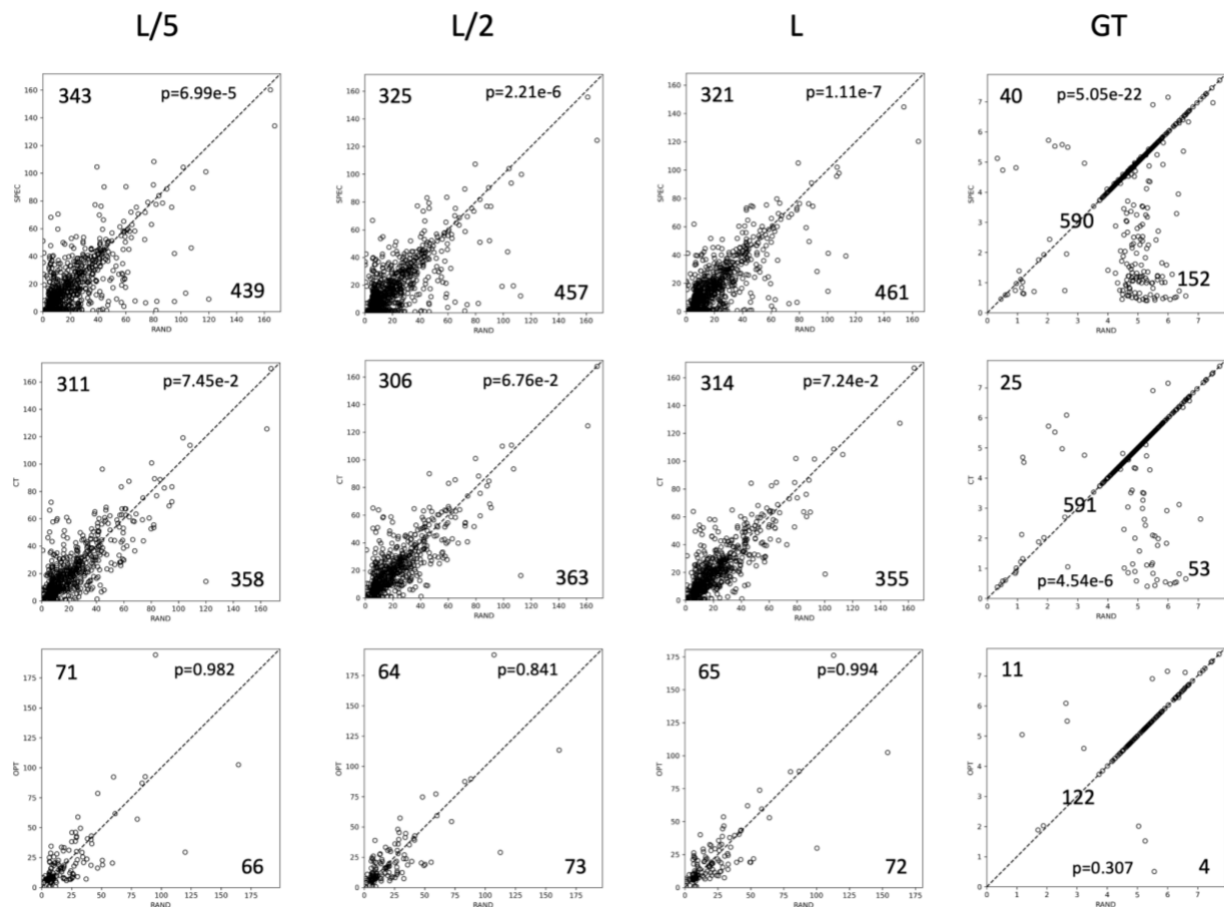


**Figure 4.2** Comparisons of the interchain distance prediction mean absolute error (MAE) for each of the tested MSA pairing schemes against the random baseline. Rows correspond to each pairing between the three programs and randomness, columns correspond to different amounts of residue pairs considered in each point's MAE (where L is the sum of the two chain's lengths); the ground truth-based MAE is labeled as "GT". Each plot is marked with the total number of points above and below (and on for the GT column) the dotted line and a p-value calculated by a two-sided Wilcoxon signed-rank test.

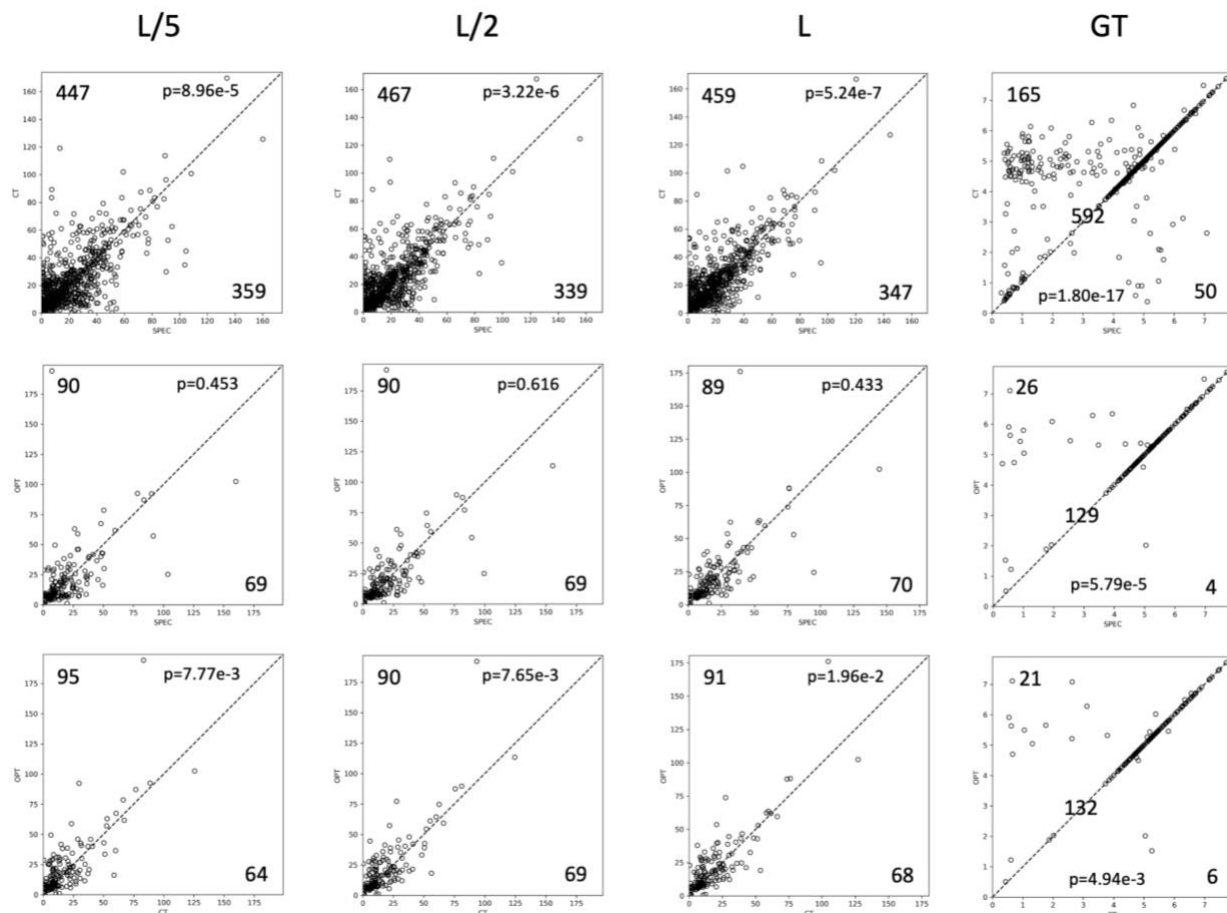


**Figure 4.3** Comparisons of the interchain distance prediction mean absolute error (MAE) for each of the tested MSA pairing schemes. Rows correspond to each pairing between the three programs, columns correspond to different amounts of residue pairs considered in each point's MAE (where L is the sum of the two chain's lengths); the ground truth-based MAE is labeled as "GT". Each plot is marked with the total number of points above and below (and on for the GT column) the dotted line and a p-value calculated by a two-sided Wilcoxon signed-rank test.

#### 4.8 Tables

|                            | L/5   | L/2   | L     | GT   |
|----------------------------|-------|-------|-------|------|
| <b>AlphaFold2 (965)</b>    | 23.09 | 22.92 | 22.68 | 4.35 |
| <b>trRosetta (960)</b>     | 36.99 | 33.23 | 31.03 | 4.35 |
| <b>DeepPotential (949)</b> | 37.07 | 33.62 | 31.50 | 4.34 |

**Table 4.1** Average MAE of each benchmark program for the top-L/5, L/2, and L distances predicted (where L is the sum of the two chain lengths). Ground truth-based MAEs are labeled as "GT". The total number of completed predictions considered in the average is indicated in parentheses (GT contains one additional point).

|                   | <b>L/5</b> | <b>L/2</b> | <b>L</b> | <b>GT</b> |
|-------------------|------------|------------|----------|-----------|
| <b>SPEC (965)</b> | 19.56      | 19.36      | 19.21    | 4.27      |
| <b>OPT (159)</b>  | 20.58      | 20.63      | 20.42    | 5.21      |
| <b>CT (808)</b>   | 21.53      | 21.65      | 21.78    | 4.66      |
| <b>RAND (783)</b> | 21.44      | 21.53      | 21.64    | 4.79      |

**Table 4.2** Average MAE of each MSA pairing strategy for the top-L/5, L/2, and L distances predicted by AlphaFold2 (where L is the sum of the two chain lengths). Ground truth-based MAEs are labeled as “GT”. The total number of completed predictions considered in the average is indicated in parentheses (GT contains one additional point for SPEC and CT).

## 4.9 References

1. Li Y, Zhang C, Bell EW, Yu DJ, Zhang Y. Ensembling multiple raw coevolutionary features with deep residual neural networks for contact-map prediction in CASP13. *Proteins*. 2019;87: 1082–1091. doi:10.1002/prot.25798
2. Li Y, Hu J, Zhang C, Yu D-J, Zhang Y. ResPRE: high-accuracy protein contact prediction by coupling precision matrix with deep residual neural networks. Valencia A, editor. *Bioinformatics*. 2019 [cited 30 Jul 2019]. doi:10.1093/bioinformatics/btz291
3. Baek M, DiMaio F, Anishchenko I, Dauparas J, Ovchinnikov S, Rie Lee G, et al. Accurate prediction of protein structures and interactions using a three-track neural network Downloaded from. Nick V Grishin. 2021;10: 22. doi:10.1126/science.abj8754
4. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Highly accurate protein structure prediction with AlphaFold. *Nat* 2021 5967873. 2021;596: 583–589. doi:10.1038/s41586-021-03819-2
5. Pozzati G, Zhu W, Bassot C, Lamb J, Kundrotas P, Elofsson A. Limits and potential of combined folding and docking. *Bioinformatics*. 2022;38: 954–961. doi:10.1093/BIOINFORMATICS/BTAB760
6. Yang J, Anishchenko I, Park H, Peng Z, Ovchinnikov S, Baker D. Improved protein structure prediction using predicted interresidue orientations. *Proc Natl Acad Sci U S A*. 2020;117: 1496–1503. doi:10.1073/pnas.1914677117
7. Evans R, O’Neill M, Pritzel A, Antropova N, Senior A, Green T, et al. Protein complex prediction with AlphaFold-Multimer. *bioRxiv*. 2021; 2021.10.04.463034. doi:10.1101/2021.10.04.463034
8. Zhou T-M, Wang S, Xu J. Deep learning reveals many more inter-protein residue-residue contacts than direct coupling analysis. *bioRxiv*. 2018 [cited 14 Oct 2022]. doi:10.1101/240754
9. Cong Q, Anishchenko I, Ovchinnikov S, Baker D. Protein interaction networks revealed by proteome coevolution. *Science*. 2019;365: 185–189. doi:10.1126/science.aaw6718
10. Li Y, Zhang C, Yu DJ, Zhang Y. Deep learning geometrical potential for high-accuracy ab initio protein structure prediction. *iScience*. 2022;25: 104425. doi:10.1016/j.isci.2022.104425
11. Bell EW, Schwartz JH, Freddolino PL, Zhang Y. PEPPI: Whole-proteome Protein-protein Interaction Prediction through Structure and Sequence Similarity, Functional Association, and Machine Learning. *J Mol Biol*. 2022; 167530. doi:10.1016/j.jmb.2022.167530

12. Zhang C, Zheng W, Mortuza SM, Li Y, Zhang Y. DeepMSA: Constructing deep multiple sequence alignment to improve contact prediction and fold-recognition for distant-homology proteins. *Bioinformatics*. 2020;36: 2105–2112. doi:10.1093/bioinformatics/btz863
13. Shen J, Zhang J, Luo X, Zhu W, Yu K, Chen K, et al. Predicting protein-protein interactions based only on sequences information. *Proc Natl Acad Sci U S A*. 2007;104: 4337–4341. doi:10.1073/pnas.0607879104
14. Steinegger M, Meier M, Mirdita M, Vöhringer H, Haunsberger SJ, Söding J. HH-suite3 for fast remote homology detection and deep protein annotation. *BMC Bioinformatics*. 2019;20: 1–15. doi:10.1186/s12859-019-3019-7

## Chapter 5 Conclusion and Future Directions

### 5.1 Experimental validation of computational approaches

Through the course of this thesis, we have presented several protein interaction networks predicted through computational means. While we have demonstrated that these networks overlap with existing literature, the most impactful portion of these networks are those which do not recapitulate what is already known. However, as is the case with any predictive method, some amount of false positives are expected to appear, and in fact, it is known that our methods tend to have difficulty distinguishing true interactions from highly homologous non-interactions. Therefore, for any novel interactions to be considered meaningful, they should be confirmed by experimental evidence.

One experimental validation project currently in progress is being performed by collaborators at the University of Michigan, where they intend to perform separate experimental determinations of the *E. coli* Nissle and *L. rhamnosus* GG to human gut epithelium protein-protein interaction networks through cross-linking mass spectrometry (XL-MS). The aim of this experimental pipeline is to capture any pairs of proteins which come into physical contact during bacterial colonization of the gut by chemically cross-linking lysine residues of any proteins in proximity to each other and identifying which proteins were cross-linked by mass spectrometry. Our XL-MS experiments will be particularly high precision due to the use of the DC4 crosslinker.[1] In this collaboration, the probiotic bacteria will be co-cultured with human enteroid monolayers and chemically crosslinked by the lab of Dr. Vincent Young. These cross-

linked samples will then be digested and analyzed by mass spectrometry by the lab of Dr. Philip Andrews. Finally, the mass spectra will be analyzed and compiled into a protein-protein interaction network by the lab of Dr. Peter Freddolino. As a compliment to these PPI experiments, transposon libraries of the bacterial strains will be constructed to determine the importance of each gene to gut colonization.

While orthogonal whole-proteome experimental methods are useful for creating a consensus network between the two approaches, the more common approach is to pick particular interactions of interest and analyze those pairs using small scale biochemical assays. For example, the PrePPI database[2] has had several studies confirm its interaction predictions through experimental means, interactions between Dclk1 and Kras,[3] and BRD4 and AMPK.[4] However, the experimental validation is not the endpoint of these analyses: the former study used the Dclk1-Kras interaction to functionally characterize Dclk1 in pancreatic cancer, and the latter study designed an inhibitor with the goal of inducing the BRD4-AMPK interaction. Such experiments are the intended purpose of predicting PPI networks, as they motivate deeper study into the specifics of the biology captured within the network. With respect to our study, we provide several avenues for experimental investigation, such as investigating the interaction between arylsulfatases in *E. coli* Nissle or whether any of the *L. rhamnosus* GG solute binding proteins modulate host cell signaling.

## **5.2 Protein dimer structure modeling**

The identification of PPI networks represents an important step towards understanding cellular function, but it is certainly not the last step towards this goal. Even if we had a hypothetically complete list of interactions occurring in a cell, this information tells us nothing about the nature and context of these interactions. A point of entry to begin interrogating this

nature is to model the interaction structure, as this model can begin to provide functional details of the interaction. While early approaches to predict quaternary structures involved docking known structures[5,6] or modeling the dimeric sequences from structural templates,[7,8] complex modeling has become the target of many state-of-the-art deep learning approaches, including AlphaFold Multimer,[9] RoseTTAFold,[10] and ColabFold.[11] Representative of this shift in focus, the most recent Critical Assessment of Techniques for Protein Structure Prediction, CASP15,[12] presented 47 multimeric targets, compared to the previous CASP14's 29 multimeric targets.[13] Considering that most of these structure predictors assume that input protein pairs are interacting, the interaction networks derived by PPI prediction can be used by these pipelines to construct structural models of not only the whole proteome, but also the whole interactome, by providing essential information on which pairs of proteins should have complex structures predicted.

One enhancement of a PPI network that is greatly facilitated by a structural model of the interactome is the estimation of protein binding affinity. By scoring predicted complex structures using energy functions developed for protein-protein docking, one can begin to distinguish between high affinity obligate complexes and more transient interactions. In addition, this type of scoring is much more sensitive to point mutations than standard PPI prediction, so binding affinity analysis could be used in combination with predicted PPI networks to determine how a given protein's binding partners, and by extension its function, will change given a single residue mutation. However, docking energy functions still demonstrate limited ability to replicate experimentally determined binding affinity,[14,15] and therefore, this field requires further development before it is utilized in proteome-wide contexts. It is possible that in the near future, a feedback loop between PPI predictions and increasingly accurate

protein-protein binding free energy calculations will allow refinement and additional annotation of PPI networks through purely computational approaches.

### 5.3 Multimeric and multi-omic interaction prediction

Many protein interaction prediction methods, including those presented in this thesis, build interaction networks through classification of individual protein pairs. However, while protein dimers are certainly important *in vivo*, a large portion of the protein-protein interaction space is occupied by protein complexes consisting of more than two protein chains. Theoretically, binary protein classifiers should be able to retrieve these complexes by predicting interactions between all the partners of the complex, but in reality, predicting multi-component complexes more often poses a difficult challenge for PPI prediction pipelines due to the necessity of all members being present to properly model the interaction.

A unique subclass of the struggles regarding multimeric PPI prediction are those regarding homo-multimers. A competent two-entity PPI classifier will certainly detect that a protein chain can interact with itself but lacks the ability to distinguish between oligomeric states and higher-order assemblies of the protein. Many programs are able to construct models of higher-order oligomers,[16–18] but the stoichiometry of these complexes must be provided. The degree to which a protein oligomerizes can have great impacts on the protein's function, and thus being able to predict these details is essential to successful description of the functional environment of the cell. As an example, the *E. coli* Leucine-responsive Regulatory Protein (Lrp) predominantly assumes an homo-octameric state in normal biological conditions (i.e., relative abundance of leucine), but is also capable of forming a homo-hexadecamer in starvation conditions (i.e., the absence of leucine). The transcription of operons regulated by Lrp is greatly impacted by the oligomerization state of Lrp, and therefore, this presents a mechanism by which

Lrp exerts leucine-sensitive transcriptional control over the *E. coli* genome.[19] The entire oligomerization-based system of transcriptional control by Lrp is neglected when only asking whether a protein can interact with itself; a multitude of similar examples exist throughout biology.

Finally, many functional complexes are not solely comprised of protein chains and in fact proteins will only confer their function when in complex with other cofactors or biomolecules. The most apparent example of this is the ribosome, which is an enormous complex of approximately equal parts rRNA and protein. Therefore, when describing the biology of such complexes, it is not sufficient to only describe the protein-protein interactions; one must also consider the RNA-RNA and protein-RNA interactions. While the prediction of protein-RNA and RNA-RNA interactions is analogous to PPI prediction, RNA tertiary structure prediction lags behind protein structure prediction, largely due to the relative lack of solved RNA structures (there are only ~1,600 RNA-only structures in the PDB).[20] Therefore, many of the structural features which have demonstrated to be useful for determining PPIs are not available for RNAs. RNA-protein interaction prediction faces unique difficulty due to the inhomogeneity of the binding entities; most predictors consider the problem from a protein-centric or RNA-centric view rather than incorporating both types of information.[21]

## **5.4 Final Conclusion**

In this thesis, we have presented a suite of tools which interrogate the construction and analysis of protein-protein interaction networks. The first of these tools was a more traditional approach based on sequence and structure homology to previously observed interactions, along with a combination of functional association and machine learning features. This tool allowed us to predict PPI networks for several cross-species biological systems which, when coupled with

our virtual reality-based network visualization software FALCON, provided novel biological insights. These insights into the colonization of the gut by several strains of probiotic bacteria warranted further experimental validation, both from orthogonal whole-proteome methods and small-scale investigations of specific interactions. However, we found that our traditional approach was outperforming deep learning approaches in species-agnostic direct interaction prediction tasks, and so we sought to improve these methods as they showed promise for becoming the state-of-the-art in PPI prediction. Through our exploration of this space, we presented a set of novel MSA pairing methods, yet preliminarily determined that the method of pairing did not impact overall performance. This particular result raised points for further investigation, such as the construction of a deep learning-based method. Regardless, the improvements and contributions we have made to this field have furthered our ability to understand the functional environment of any cellular system which has a known proteome, and as a result, we have improved studies which use this information to diagnose and develop therapeutics for various diseases.

## 5.5 References

1. Clifford-Nunn B, Showalter HDH, Andrews PC. Quaternary diamines as mass spectrometry cleavable crosslinkers for protein interactions. *J Am Soc Mass Spectrom.* 2012;23: 201–212. doi:10.1007/s13361-011-0288-4
2. Zhang QC, Petrey D, Deng L, Qiang L, Shi Y, Thu CA, et al. Structure-based prediction of protein-protein interactions on a genome-wide scale. *Nature.* 2012;490: 556–560. doi:10.1038/nature11503
3. Westphalen CB, Takemoto Y, Tanaka T, Macchini M, Jiang Z, Renz BW, et al. Dclk1 Defines Quiescent Pancreatic Progenitors that Promote Injury-Induced Regeneration and Tumorigenesis. *Cell Stem Cell.* 2016;18: 441–455. doi:10.1016/j.stem.2016.03.016
4. Ouyang L, Zhang L, Liu J, Fu L, Yao D, Zhao Y, et al. Discovery of a Small-Molecule Bromodomain-Containing Protein 4 (BRD4) Inhibitor That Induces AMP-Activated Protein Kinase-Modulated Autophagy-Associated Cell Death in Breast Cancer. *J Med Chem.* 2017;60: 9990–10012. doi:10.1021/acs.jmedchem.7b00275
5. Dominguez C, Boelens R, Bonvin AMJJ. HADDOCK: A protein-protein docking approach based on biochemical or biophysical information. *J Am Chem Soc.* 2003;125:

- 1731–1737. doi:10.1021/ja026939x
6. Gray JJ, Moughon S, Wang C, Schueler-Furman O, Kuhlman B, Rohl CA, et al. Protein-protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations. *J Mol Biol.* 2003;331: 281–299. doi:10.1016/S0022-2836(03)00670-3
7. Lu L, Arakaki AK, Lu H, Skolnick J. Multimeric threading-based prediction of protein-protein interactions on a genomic scale: Application to the *saccharomyces cerevisiae* proteome. *Genome Res.* 2003;13: 1146–1154. doi:10.1101/gr.1145203
8. Guerler A, Govindarajoo B, Zhang Y. Mapping monomeric threading to protein-protein structure prediction. *J Chem Inf Model.* 2013;53: 717–725. doi:10.1021/ci300579r
9. Evans R, O'Neill M, Pritzel A, Antropova N, Senior A, Green T, et al. Protein complex prediction with AlphaFold-Multimer. *bioRxiv.* 2021; 2021.10.04.463034. doi:10.1101/2021.10.04.463034
10. Baek M, DiMaio F, Anishchenko I, Dauparas J, Ovchinnikov S, Rie Lee G, et al. Accurate prediction of protein structures and interactions using a three-track neural network Downloaded from. Nick V Grishin. 2021;10: 22. doi:10.1126/science.abj8754
11. Mirdita M, Schütze K, Moriwaki Y, Heo L, Ovchinnikov S, Steinegger M. ColabFold: making protein folding accessible to all. *Nat Methods.* 2022;19: 679–682. doi:10.1038/s41592-022-01488-1
12. Home - CASP15. [cited 18 Oct 2022]. Available: <https://predictioncenter.org/casp15/index.cgi>
13. Kryshchuk A, Schwede T, Topf M, Fidelis K, Moult J. Critical assessment of methods of protein structure prediction (CASP)—Round XIV. *Proteins.* 2021;89: 1607–1617. doi:10.1002/prot.26237
14. Kastitis PL, Bonvin AMJJ. Are scoring functions in protein-protein docking ready to predict interactomes? Clues from a novel binding affinity benchmark. *J Proteome Res.* 2010;9: 2216–2225. doi:10.1021/pr9009854
15. Siebenmorgen T, Zacharias M. Computational prediction of protein–protein binding affinities. *Wiley Interdisciplinary Reviews: Computational Molecular Science.* Blackwell Publishing Inc.; 2020. p. e1448. doi:10.1002/wcms.1448
16. Guo Z, Liu J, Cheng J. Prediction of inter-chain distance maps of protein complexes with 2D attention-based deep neural networks. *bioRxiv.* 2022 [cited 18 Oct 2022]. doi:10.1101/2022.06.19.496734
17. Yan Y, Huang SY. Accurate prediction of inter-protein residue-residue contacts for homooligomeric protein complexes. *Brief Bioinform.* 2021;22: 1–13. doi:10.1093/bib/bbab038
18. Roy RS, Quadir F, Soltanikazemi E, Cheng J. A deep dilated convolutional residual network for predicting interchain contacts of protein homodimers. *Bioinformatics.* 2022;38: 1904–1910. doi:10.1093/bioinformatics/btac063
19. Chen S, Calvo JM. Leucine-induced dissociation of *Escherichia coli* Lrp hexadecamers to octamers. *J Mol Biol.* 2002;318: 1031–1042. doi:10.1016/S0022-2836(02)00187-0
20. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, et al. The Protein Data Bank. *Nucleic Acids Res.* 2000;28: 235–242. doi:10.1093/nar/28.1.235
21. Wei J, Chen S, Zong L, Gao X, Li Y. Protein–RNA interaction prediction with deep learning: structure matters. *Brief Bioinform.* 2022;23. doi:10.1093/bib/bbab540

## **Appendix A: DockRMSD: an open-source tool for atom mapping and RMSD calculation of symmetric molecules through graph isomorphism**

This appendix was adapted from the manuscript Eric W. Bell, Yang Zhang.

“DockRMSD: an open-source tool for atom mapping and RMSD calculation of symmetric molecules through graph isomorphism” (2019). Journal of Cheminformatics 11, 40.[1] This article was distributed via the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>) and was modified from its original form to match the formatting of this dissertation.

### **A.1 Abstract**

Comparison of ligand poses generated by protein–ligand docking programs has often been carried out with the assumption of direct atomic correspondence between ligand structures. However, this correspondence is not necessarily chemically relevant for symmetric molecules and can lead to an artificial inflation of ligand pose distance metrics, particularly those that depend on receptor superposition (rather than ligand superposition), such as docking root mean square deviation (RMSD). Several of the commonly-used RMSD calculation algorithms that correct for molecular symmetry do not take into account the bonding structure of molecules and can therefore result in non-physical atomic mapping. Here, we present DockRMSD, a docking pose distance calculator that converts the symmetry correction to a graph isomorphism searching problem, in which the optimal atomic mapping and RMSD calculation are performed by an exhaustive and fast matching search of all isomorphisms of the ligand structure graph. We show

through evaluation of docking poses generated by AutoDock Vina on the CSAR Hi-Q set that DockRMSD is capable of deterministically identifying the minimum symmetry-corrected RMSD and is able to do so without significant loss of computational efficiency compared to other methods. The open-source DockRMSD program can be conveniently integrated with various docking pipelines to assist with accurate atomic mapping and RMSD calculations, which can therefore help improve docking performance, especially for ligand molecules with complicated structural symmetry.

## A.2 Introduction

Computer-aided drug design, in particular protein–ligand docking, has brought about the discovery of many biologically active drugs.[2,3] In many protein–ligand docking programs, a flexible small molecule structure is docked in a rigid protein receptor structure in order to find the optimal binding conformation and affinity of the small molecule within the protein binding pocket. Since the ability of these programs to accurately assess binding affinity is dependent on their ability to find the optimal conformation of the ligand in the protein binding pocket, docking programs are often benchmarked by their ability to reproduce the native binding pose of a ligand from a protein–ligand complex crystal structure. A common metric used to evaluate distance between the predicted pose and the native pose, given a superposition of their protein receptor structures, is the root mean square deviation (RMSD) between their respective atoms (Eq. 1):

$$RMSD = \sqrt{\frac{1}{N} \sum_{i=1}^N d_i^2} \quad (1)$$

where  $N$  is the number of atoms in the ligand, and  $d_i$  is the Euclidean distance between the  $i$ th pair of *corresponding* atoms.

Docking RMSD can be most naïvely calculated with the assumption of direct atomic correspondence, or in other words, the assumption that the atomic labels between ligand structures in the given structure files are ordered and should remain static in the docking process. This assumption holds for asymmetric molecules like caffeine (**Figure A.1a**), but this correspondence is not always practically relevant for molecules with symmetric functional groups (e.g., ibuprofen, **Figure A.1b**) or whole-molecule symmetry (e.g., the pyrrolidine-based inhibitor of HIV-1 protease[4] in **Figure A.1c**), as they can give rise to binding poses that are identical in terms of chemistry, but not in terms of correspondence. Here, ibuprofen and HIV-1 protease pyrrolidine-based inhibitor have been chosen as illustrative examples, although there are various other molecules with symmetric structures in which naïve correspondence can result in false inflation of RMSD (the inhibitor BEA403,[5] c-di-GMP,[6] etc.). For example, if one were to perfectly overlap two benzene molecules, their docking RMSD would have a value of zero. If one were to then rotate one molecule along one of its axes of symmetry until the two structures overlapped perfectly again, their docking RMSD should be zero due to the chemical identity of the overlap; since the overlapping atoms are differently labeled between the two molecules in this example, naïve docking RMSD would have a nonzero value. Therefore, molecular symmetry needs to be taken into account in order to derive an accurate docking RMSD value.

Several docking programs have implemented docking RMSD modules to accommodate ligand symmetry. AutoDock Vina[7] was one of the first to implement symmetry correction in docking RMSD calculation, providing a module that creates correspondence by mapping each atom of one pose to the closest atom of the same type from the other pose. However, this method allows the potential for atoms that are close between the two structures to be used repeatedly and atoms that are distant to not be used at all. In response to this, Allen and Rizzo[8] implemented

their own docking RMSD calculator in DOCK6[9] which presents atomic correspondence mapping as a cost-minimization assignment problem, solved by using the Hungarian algorithm.[10,11] However, considering the mapping problem in this way ignores the bonding structure of the ligand, and can potentially provide nonphysical assignments (**Figure A.1d**) and docking RMSD values that are lower than what should be physically possible. Several other docking programs, such as GOLD,[12] AmberTools,[13] and Glide[14,15] also contain modules that calculate symmetry-corrected RMSD, but these modules generally do not publicly offer thoroughly detailed explanations of their symmetry correction algorithms and demand that the user install a much larger package to calculate symmetry corrected RMSD. Finally, OpenBabel[16] contains a C++ open source tool, *obrms*, that considers symmetry correction as a graph isomorphism problem, solved by the VF2 algorithm,[17] but also currently requires that the user install the entirety of OpenBabel to use this tool. Docking RMSD calculated by these modules is distinct from conformational distance metrics calculated by programs such as LS-align[18] and RDKit,[19] as these metrics are based on a superposition of the ligand structures themselves, not the receptor on which they are docked. Such a superposition is inappropriate for evaluation of docking poses due to the lack of consideration of the position and orientation of the ligand relative to the receptor; it is more appropriate for purely cheminformatic problems, such as ligand structural similarity comparisons. Therefore, there exists a need for a universal docking RMSD calculation module that properly considers molecular symmetry and does so with a clear, detailed description of its methodology.

Here we propose a new, open-source module, DockRMSD, to solve the atom mapping issue for symmetric molecular structures through graph isomorphism, where the optimal docking RMSD is calculated by searching through a pruned state space of all isomorphic mappings

between two molecular structures. Source code in C, compiled binaries, and a web server implementation of DockRMSD are made freely available at the DockRMSD web site.[20]

### A.3 Implementation

A general overview of the DockRMSD algorithm is presented in **Figure A.2**. To begin, the user provides a pair of structure files in MOL2 format, each containing a specific pose of the same ligand. The first file is arbitrarily defined as the “query” structure and the second as the “template” structure, for convenience of description. The elements of the heavy (non-H) atoms present in each structure, the coordinates of those atoms, and the bonding network between the pairs of atoms are read from the structure files. Subsequently, the atom and bond sets are compared in order to ensure that the two structures are of the same ligand molecule. Bonds are represented by a symmetric two-dimensional array which contains a string corresponding to bond type (single = “1”, double = “2”, aromatic = “ar”, etc.) between bonded atoms  $i$  and  $j$  at array position  $[i, j]$ , and contains empty strings otherwise. If the bond types do not agree between the two files, the bond network is stripped of bond types, preserving only which atoms are bonded.

Once the ligand structural information has been extracted, the next step is to determine the set of template atoms to which each query atom is chemically identical, referred to as the atom identity search. For each atom of the query structure, all atoms of the template structure of the same element are initially considered to be candidate mapping partners. Then, the set of atoms that the query atom is bonded to, as well as the bond types between them, is evaluated against the set of atoms and bonds present for each candidate mapping partner in the template; candidate template atoms are eliminated if their bonding structure does not match the query atom. This process is repeated, checking for identity between not only their set of bonded

neighbor atoms, but the neighbors of those neighbor atoms as well, once again removing candidate atoms if the sets are not identical. A deeper search involving further neighbor atoms was attempted, but it was found that including more neighbors ultimately did not change the final optimal correspondence. Therefore, the identity search stops at this neighbor atom depth in order to minimize unnecessary neighbor set comparisons and optimize runtime. If more than one candidate remains, there is likely more than one atom in the template that is chemically identical to the query atom, meaning that the ligand has some degree of symmetry. Once the atom identity search is complete, each query atom will have a set of template atoms that are chemically equivalent to that query atom. For a completely asymmetric molecule (**Figure A.1a**), each query atom will only have one corresponding template atom. Therefore, calculating the optimal RMSD for these asymmetric molecules is a simple task of matching each query atom to its respective template atom and returning the RMSD calculated from this correspondence. However, for symmetric molecules, one must search through all possible assignments of template atoms in order to find the mapping whose RMSD is minimal. The putative computational expense of this search is calculated as the total number of possible mappings, i.e., the product of each query atoms' candidate atom set length.

In order to find the deterministically optimal mapping between query and template atoms, an exhaustive assignment search reminiscent of the VF2 algorithm[17] coupled with Dead-End Elimination (DEE)[21] is implemented. In this procedure, query atoms are iteratively assigned the template label that provides the smallest squared interpose distance and can feasibly added to the existing assignments. The first of these feasibility criteria is if the candidate template atom that is being assigned has already been assigned to a previous query atom, this assignment is disallowed. This restriction ensures that all mappings are one-to-one such that no template atom

is mapped to more than one query atom. Second, if the query atom that is currently being mapped is bonded to already mapped atoms, the template bonding network is checked to ensure that a bond also exists in the template between the labels given to those atoms from the query. If the bonds being formed by query atom assignment are not supposed to be formed according to the template, the proposed assignment is not feasible. Finally, the last feasibility criterion is DEE, which ceases assignment of a particular atom if all subsequent feasible assignments would result in an RMSD larger than the smallest heretofore observed RMSD (which is infinity if no RMSD has yet been observed). The query atoms are mapped in order of number of possible template labels (smallest first), then number of bonds to already mapped query atoms (largest first), and finally, the order in which they appear in the query file (smallest first). Once all query atoms have been assigned to template atoms, this correspondence is used to calculate RMSD. The minimum RMSD for all mappings and the mapping that gave rise to that RMSD are then printed by the program. In addition, the number of possible mappings is printed.

## **A.4 Results and discussion**

### ***A.4.1 Docking conformation dataset and generation***

To evaluate DockRMSD's symmetry correction and the reliability of the greedy search heuristic, we generated docking conformations based on the CSAR Hi-Q protein-ligand dataset.[22] This dataset contains 343 protein structures with manually refined binding pockets, each in complex with their respective ligand, where the docking decoy conformations have been generated by ourselves using the AutoDock Vina program.[7] For each protein–ligand pair, the native ligand structure was removed, conformationally randomized using OpenBabel,[16] and redocked into the binding pocket using AutoDock Vina.[7] The generation of input PDBQT files for docking and the output file conversion from PDBQT to MOL2 was performed by

OpenBabel. Docking RMSD was calculated between all 10 possible pairwise combinations of the top five poses generated from a single re-docking experiment, leading to a total of 3430 RMSD calculations (10 per protein–ligand pair, 343 protein–ligand pairs in total). All 3430 calculations were performed using a list of different programs on a Red Hat Enterprise Linux machine with an Intel i5-4590 CPU @ 3.30 GHz. The average total walltime for all 3430 RMSD calculations was  $4.8 \pm 0.7$  s,  $5.3 \pm 0.9$  s, and  $60.1 \pm 0.1$  s for DockRMSD, naïve RMSD, and *obrms*, respectively (see section A.4.3 for a more detailed runtime analysis).

Here, naïve RMSD calculations relative to the native crystal structure pose were not calculated because the AutoDock Vina ligand preparation process removes direct atomic correspondence between the redocked ligand and the native ligand. AutoDock Vina re-orders the atoms of the ligand according to the ligand’s torsional tree, and therefore, all Vina poses have direct correspondence with each other, but not with the original native ligand structure. Therefore, only programs that can find atomic correspondence between files can be used to compare the Vina poses to the native crystal structure pose. This limitation is why the dataset used to evaluate the programs consists only of docked poses; direct correspondence cannot be drawn between the native crystal structure and Vina-generated poses. Ligand structures have been visualized using UCSF Chimera.[23]

#### ***A.4.2 Docking RMSD calculation through DockRMSD***

To examine the impact of symmetry correction in docking RMSD calculation, we compare in **Figure A.3a** the symmetry-corrected RMSD calculated from DockRMSD and the naïve RMSD which was calculated from the default atom order of the structure files. While 2109 of the 3430 cases require no symmetry correction, the remaining 1321 (38.5%) are cases where adhering to naïve RMSD artificially inflates the docking RMSD, by more than 2 Å in 54 of these

cases (**Table A.1**). The most extreme examples of this are when a ligand molecule is large and possesses a mirror plane of symmetry, and when the ligand poses roughly overlap. For these cases, determining the optimal mapping is essential because misplaced correspondence will give rise to unreasonably large interatom distances, especially when compared to the relatively small “true” RMSD. An example of a Huperzine A-based ligand of acetylcholinesterase[24] is shown in **Figure A.3b**, where the two halves of the molecule are chemically identical to another and by eye should have a relatively small RMSD value. DockRMSD’s calculation aligns with this rough assessment, calculating an RMSD value of 3.42 Å. However, due to the fact that the query is flipped relative to the template, naïve RMSD considers this reorientation an important distinction, and therefore calculates the RMSD to be 10.74 Å.

In **Figure A.4a**, we present a comparison between the RMSD of DockRMSD and that calculated by the Hungarian algorithm, which has been adopted by several established methods, such as DOCK6.[9] In the Hungarian algorithm, the mapping is generated through iterative manipulation of a cost matrix (i.e., an interatom distance matrix) such that a pattern of zero values corresponding to the optimal assignment appears. The performance of the Hungarian algorithm was evaluated using a Python implementation of the docking RMSD calculation procedure similar to what is described by Allen and Rizzo.[8] The script uses the Python Munkres package[25] to generate query-template atomic correspondence such that assignments can only be made between atoms of the same element. As explained above, the laxness of this algorithm causes it to over-optimize and generate RMSD values below what should be possible. As a result, in nearly every case analyzed, the Hungarian algorithm generated an RMSD value below the optimal answer found by DockRMSD (3269 of 3430 RMSD calculations, 95.3%;

**Table A.1).** This implies that the over-correction issue introduced by the Hungarian algorithm is not trivial.

In contrast to the comparison between DockRMSD and naïve RMSD, the largest discrepancies between DockRMSD and the Hungarian algorithm are present in near mirror-symmetric molecules whose poses overlap almost exactly. As an illustrative example, we present in **Figure A.4c** a result from the HIV-1 protease inhibitor BEA425,[26] where the poses presented look nearly identical by eye, and thus, one would anticipate the RMSD value should be low. However, this molecule is not truly symmetric due to a hydroxyl group near the center of the molecule, and therefore, the two poses are not truly chemically identical. Since the Hungarian algorithm only takes into account individual atom types and not global chemical identity, cases like these fool the algorithm into accepting regions of local correspondence at the cost of properly considering which atoms are bonded. Although the algorithm generates lower RMSD values, these values do not reflect correct correspondence of the atomic mapping derived from the ligand bonding structures.

Here, it is noted that the above RMSD calculations are performed on the AutoDock Vina docked conformations, which was chosen purely to enable the comparison of different RMSD calculation programs with direct correspondence. In fact, one of the most common applications of ligand RMSD calculation is for benchmarking experiments that evaluate a docking program's ability to produce ligand poses that closely resemble the native conformation. In such experiments, poses are typically considered “near-native” if their RMSD relative to the native pose is  $\leq 2.0$  Å. In order to examine the performance of different programs with respect to this task, the top ranked AutoDock Vina pose for each of the 343 protein–ligand pairs was compared against the crystal structure pose of the ligand as provided by the CSAR Hi-Q set using both

DockRMSD and the Hungarian algorithm, the results of which are presented in **Figure A.4b** and **Table A.2**. It was shown that in 190 of the 343 cases, the Hungarian algorithm resulted in a lower value than the optimal value as determined by DockRMSD, where 10 of them would have resulted in a false positive classification of a “near native” pose. These results demonstrate that evaluation of a docking algorithm by RMSD values using incorrect atomic correspondences can lead to artificial inflations of docking results.

#### **A.4.3 DockRMSD runtime comparison**

In order to evaluate the runtime efficiency of DockRMSD, both naïve and symmetry-corrected RMSD calculations on all 3430 pose pairs were compared to the runtimes of *obrms*. The *obrms* package is a tool from OpenBabel that calculates RMSD through solving the graph isomorphism problem using a similar algorithm relative to DockRMSD. The values calculated between *obrms* and DockRMSD (if the bond type information is not used in DockRMSD) are identical; therefore, the most poignant comparison between these two programs is to determine how quickly they respectively come to the correct answer. The results of this experiment are summarized in **Figure A.5**, with runtimes being log-transformed to more closely resemble normal distributions. As is shown, every calculation performed by DockRMSD was faster than the fastest calculation made by *obrms*, which is consistent with the statistically significant difference between their average runtimes ( $t = 310.6$ ,  $p < 10^{-20,000}$  by one-tailed paired t-test). The difference between symmetry-corrected and symmetry-uncorrected runtime is also statistically significant ( $t = 43.9$ ,  $p < 10^{-400}$  by one-tailed paired t-test), but the magnitude of mean difference between DockRMSD and *obrms* ( $1.04 \log_{10}(\text{seconds})$ ) is much larger than between symmetry-corrected and naïve runtime ( $0.21 \log_{10}(\text{seconds})$ ). This data suggests that while the

impact of symmetry correction on RMSD calculation time is observable, its impact on runtime relative to *obrms*, which performs a similar symmetry correction, is minimized.

While a good portion of this runtime difference can be attributed to the fact that *obrms* is implemented using OpenBabel's object-oriented framework and thus leads to the instantiation of more computationally intensive data structures than is necessary for this problem, DEE also contributes to the increased efficiency of DockRMSD. As an illustrative example of DEE's power, a buckminsterfullerene ( $C_{60}$ ) molecule was docked onto tRNA-Guanine Transglycosylase[27] using AutoDock Vina, and subsequently, docking RMSD was calculated between the top five poses using DockRMSD without DEE, DockRMSD with DEE, and *obrms* for runtime analysis. The choice of receptor was random and arbitrary in this experiment; docking on this receptor was only a means to generate hypothetical poses for the ligand and implies no greater biological relevance. However, the reason buckminsterfullerene molecule was chosen as the ligand is that it is one of the most highly symmetric molecules that has been observed in nature: each carbon is chemically identical to every other carbon in the molecule, leading to a total state space of  $60^{60}$  possible mappings, a greater number of mappings than there are atoms in the universe. Therefore, proper pruning of the mapping search space is essential to efficiently find the minimum RMSD for this molecule. Reflective of this, DockRMSD without DEE requires a relatively high amount of time (on average  $93.3 \pm 0.9$  ms per ligand pair) to find the optimal solution, as the only pruning done is the bond-based and duplicate criteria described in the implementation; the atom identity search provides no information due to the symmetry of buckminsterfullerene. The *obrms* tool prunes more efficiently (on average  $59.6 \pm 0.9$  ms per ligand pair) due to its direct implementation of the VF2 feasibility criteria, but still needs to enumerate through every valid mapping to find the optimal

one and thus takes longer to arrive at the optimal answer. However, since DEE prunes mappings based on their cumulative square distance, DockRMSD is able to find the optimal solution within a timeframe that rivals the runtime of *obrms* on most other molecules (on average  $8.7 \pm 0.7$  ms per ligand pair).

## A.5 Conclusion

The inability of naïve RMSD calculation to account for molecular symmetry negatively impacts how we evaluate ligand poses generated by protein–ligand docking. In the dataset we analyzed, about two out of every five ligands require some sort of symmetry correction to achieve accurate docking RMSD values, some of which demonstrated an RMSD correction of more than 2.0 Å. While several attempts have been made to address this need, implementations that find mappings without considering atomic connectivity, such as those in DOCK6 and AutoDock Vina, ultimately fail to consider properties of the ligand that are necessary to find the true optimal symmetry-corrected RMSD. While modules from commercial programs like GOLD and Glide are also capable of finding the optimal solution and are likely more convenient if the poses being evaluated were generated from these programs, users who wish to use these programs must purchase a license or install hefty software packages to perform RMSD calculations. Finally, even when compared to analogous open-source modules, such as *obrms*, DockRMSD has demonstrated much faster calculations in all cases (particularly high-symmetry cases) due to its lightweight implementation. In addition to symmetry correction, the atomic correspondence search of DockRMSD promotes easier comparison between docking programs in benchmarking studies. Ligand poses generated by several programs do not necessarily have direct atomic correspondence, and so DockRMSD could be used as a universal analysis module to ensure all programs are able to be compared and that the comparison is fair.

Despite the ability of DockRMSD to calculate symmetry-corrected RMSD, a few shortcomings of the program remain. For example, DockRMSD requires that the two molecules that are provided are the same molecule due to the atom identity search step. This could potentially be solved through implementation of maximum common substructure searching. However, if the molecule being analyzed is symmetric, a common substructure could potentially correspond to several positions in the molecule, leading to several different potential RMSD values. In addition, DockRMSD currently only evaluates ligand pose distance through docking RMSD because of the popularity of this metric. However, RMSD is far from a perfect metric, particularly because of its inability to capture the conservation of essential protein–ligand interactions that confer high binding affinity. As of now, DockRMSD does not include metrics that address these shortcomings of RMSD because they require the consideration of the protein receptor structure, but future iterations of this software could feasibly incorporate this information along with the typical RMSD calculation.

## **A.6 Acknowledgements**

We would like to thank Dr. Wallace Chan for his critique of this manuscript. Molecular graphics performed with UCSF Chimera, developed by the Resource for Biocomputing, Visualization, and Informatics at the University of California, San Francisco, with support from NIH P41-GM103311. The study is supported in part by the National Institute of General Medical Sciences [GM070449, GM083107, GM116960], National Institute of Allergy and Infectious Diseases [AI134678], and the National Science Foundation [DBI1564756].

## A.7 Figures

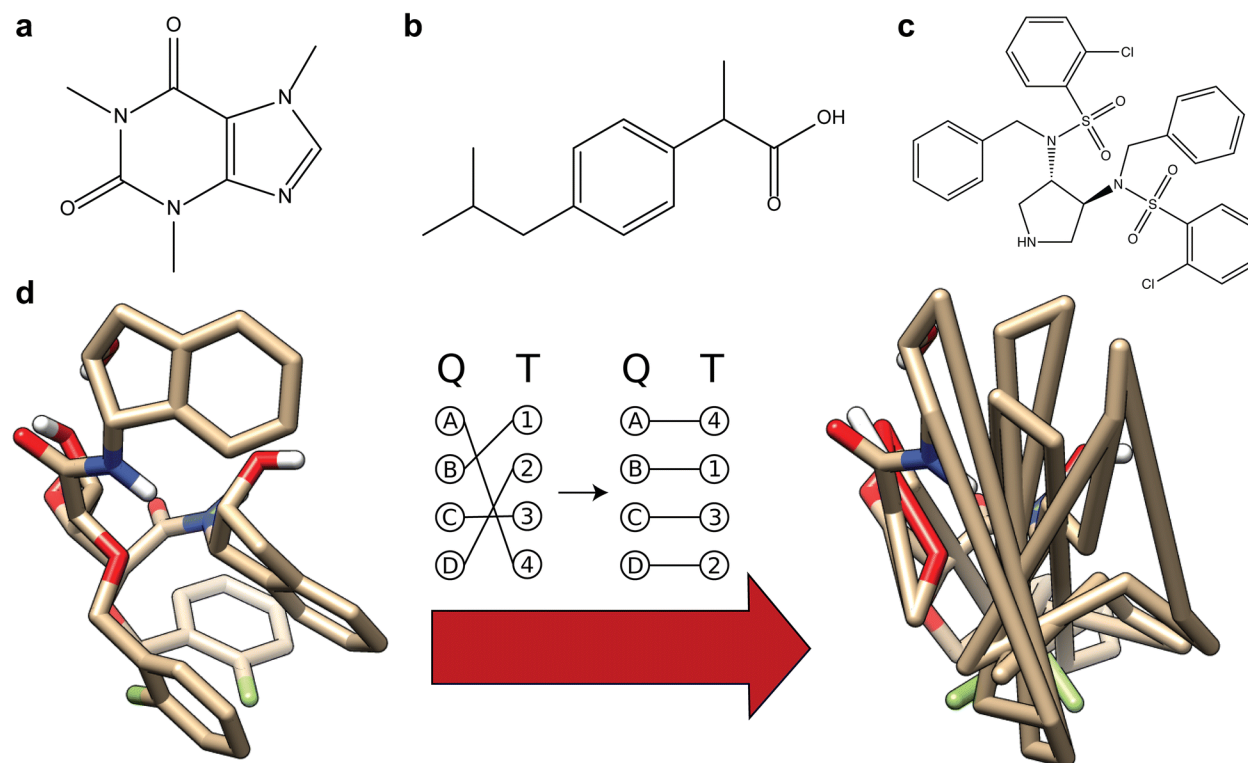


**Figure A.1** Examples of (a) an asymmetric ligand (PDB Ligand ID: CFF); (b) a slightly symmetric ligand (PDB Ligand ID: IBP); (c) a highly symmetric ligand (PDB Ligand ID: QN3). (d) An example ligand structure (left) and the resulting ligand structure when the atoms are reordered according to the optimal query-template atomic correspondence generated by the Hungarian method (right). Since the Hungarian method only takes atom type into account and not the bonds between atoms, the hypothetical molecule proposed by the Hungarian correspondence is physically impossible.

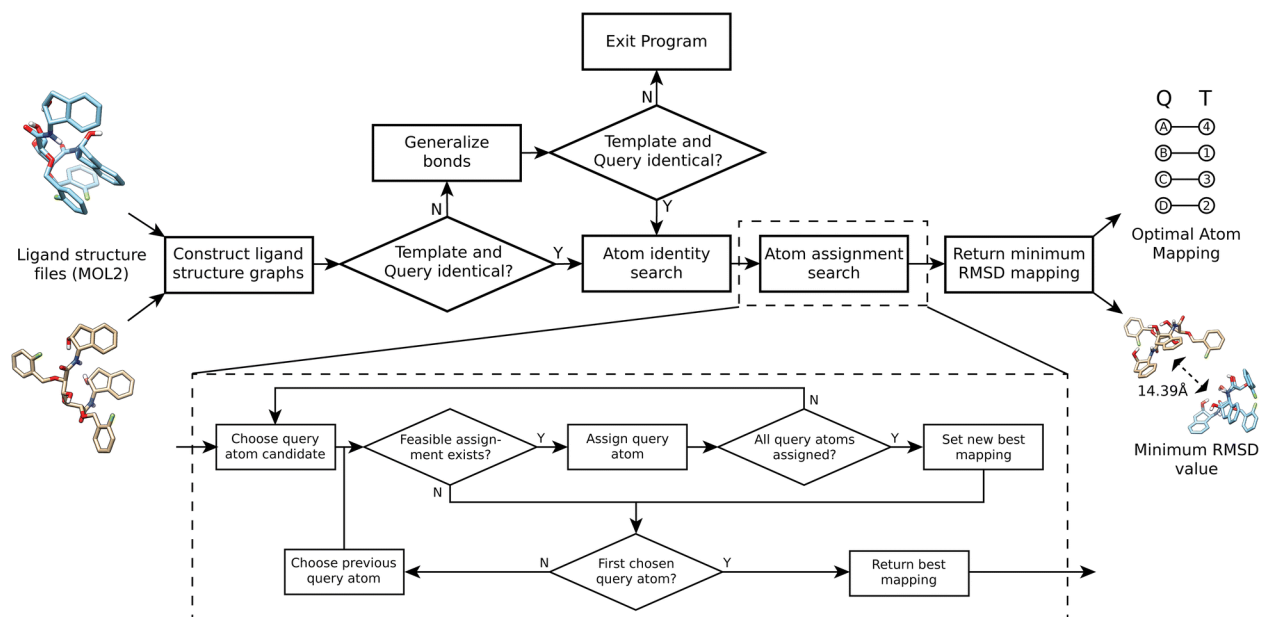


**Figure A.2** The DockRMSD algorithm. DockRMSD calculates the optimal atom mapping and RMSD value for any given pair of poses for the same ligand, input as a pair of MOL2 structure files.

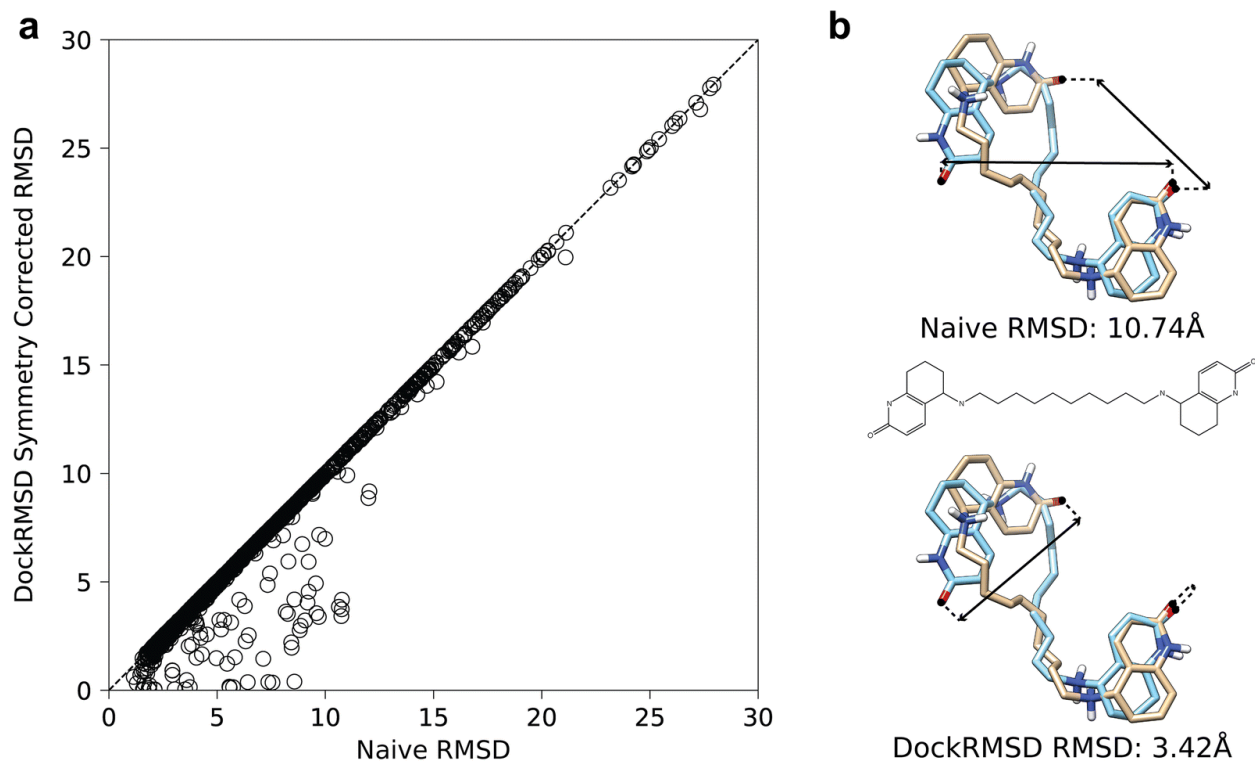


**Figure A.3** (a) Ligand RMSD calculated by DockRMSD versus that by the naïve RMSD calculations on 1321 ligand molecules with symmetric structures. (b) An example pair of poses where naïve RMSD calculation failed to provide the optimal RMSD due to molecular symmetry (Ligand PDB ID: E10; Receptor PDB ID: 1H22 [24]). Interpose correspondence between oxygen atoms is drawn to represent the source of the RMSD disagreement by different methods.

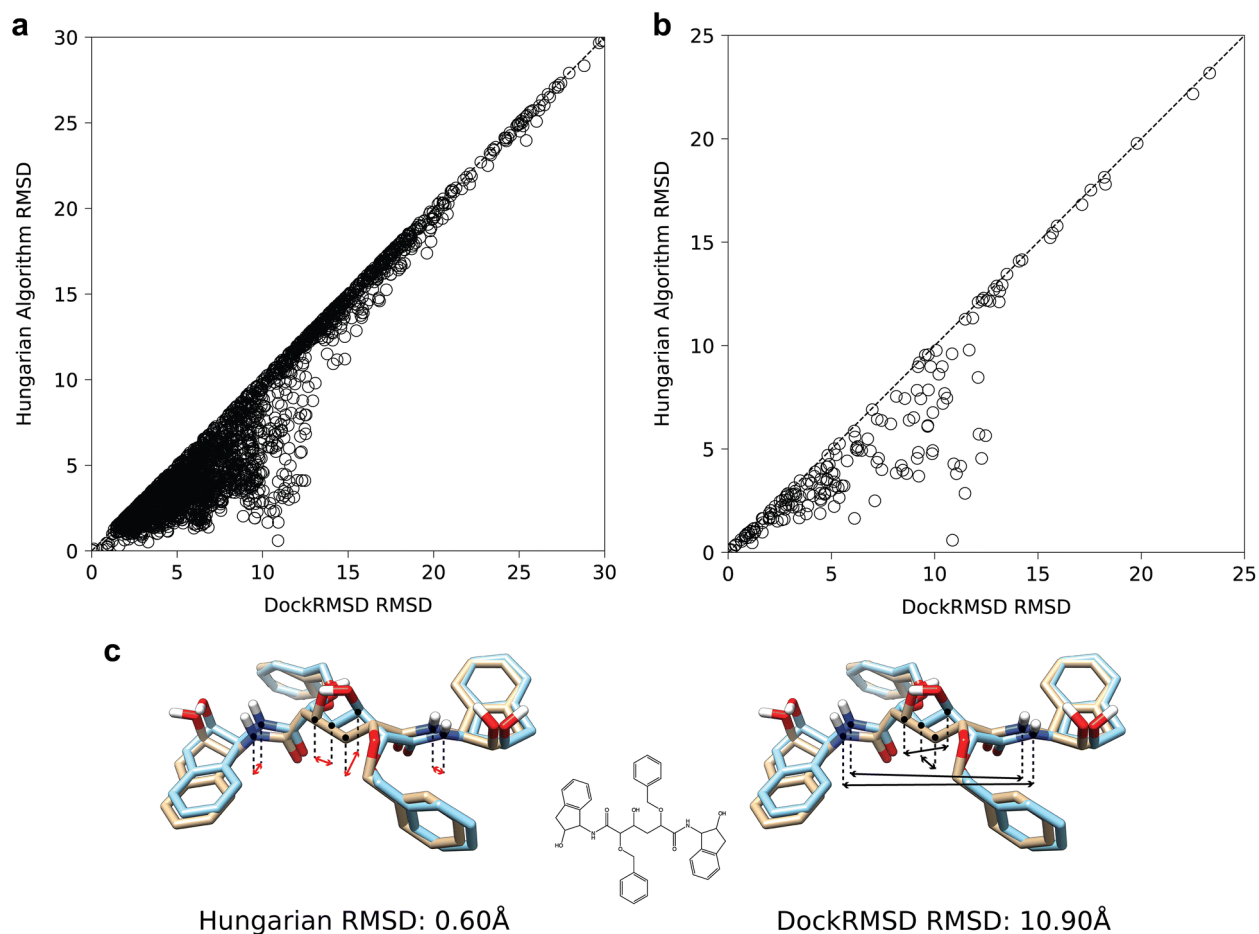


**Figure A.4** (a) Comparison of the Hungarian algorithm against DockRMSD for the 3112 molecules whose RMSD was underestimated by the Hungarian algorithm. (b) Comparison of the Hungarian algorithm against DockRMSD for the 190 molecules whose RMSD was underestimated by the Hungarian algorithm in the native ligand pose benchmark. (c) An example pair of poses where the Hungarian algorithm grossly overcorrected for symmetry due to its insensitivity to global molecular topology (Ligand PDB ID: BEG; Receptor PDB ID: 1D4I [26]). Interpose correspondence between central carbon atoms and nitrogen atoms is drawn to represent the source of the RMSD disagreement. Hungarian correspondence is drawn in red to demonstrate that the correspondence should not be allowed according to the chemical inequivalence of the atoms bonded to each atom of the pair.

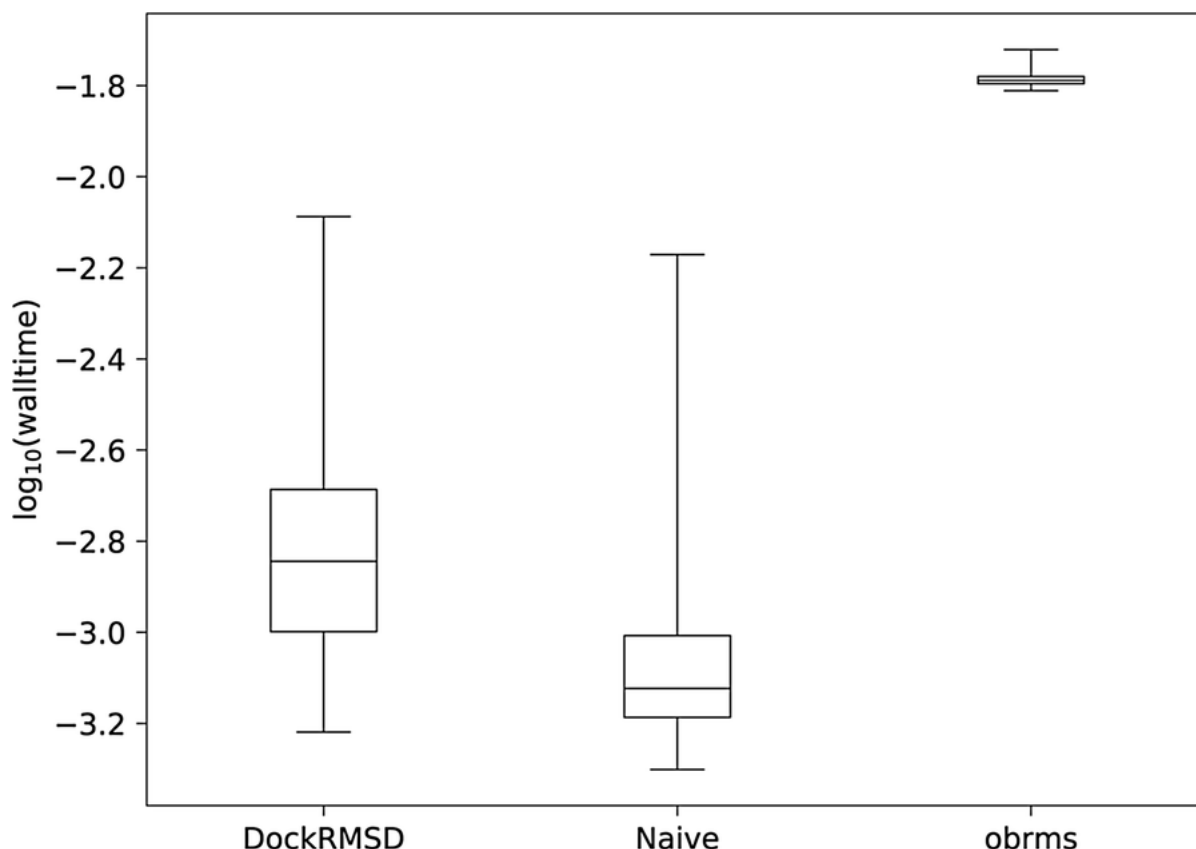


**Figure A.5** Box and whisker plots of the walltime distributions (in  $\log_{10}(\text{sec})$ ) for each of the 3430 RMSD calculations as calculated by symmetry-corrected DockRMSD, symmetry-uncorrected naïve RMSD, and symmetry-corrected obrms

## A.8 Tables

|           | N (Error = 0) | N ( $0 < \text{Error} \leq 2.0\text{\AA}$ ) | N (Error $> 2.0\text{\AA}$ ) |
|-----------|---------------|---------------------------------------------|------------------------------|
| Naïve     | 2109 (61.5%)  | 1267 (36.9%)                                | 54 (1.6%)                    |
| Hungarian | 161 (4.7%)    | 2548 (74.3%)                                | 721 (21.0%)                  |

**Table A.1** Counts of 3,430 total RMSD calculations whose error relative to the deterministic DockRMSD calculation is zero, small (nonzero but smaller than  $2.0\text{\AA}$ ), or large (greater than  $2.0\text{\AA}$ ).

|                               | Hungarian RMSD $> 2.0\text{\AA}$ | Hungarian RMSD $\leq 2.0\text{\AA}$ |
|-------------------------------|----------------------------------|-------------------------------------|
| DockRMSD $> 2.0\text{\AA}$    | 157                              | 10                                  |
| DockRMSD $\leq 2.0\text{\AA}$ | 0                                | 176                                 |

**Table A.2** A contingency table for 343 RMSD calculations between docked ligand poses and their respective native crystal structure ligand poses, calculated both by DockRMSD and the Hungarian algorithm.

## A.9 References

1. Bell EW, Zhang Y. DockRMSD: an open-source tool for atom mapping and RMSD calculation of symmetric molecules through graph isomorphism. *J Cheminform.* 2019;11: 40. doi:10.1186/s13321-019-0362-7
2. Tuccinardi T. Docking-Based Virtual Screening: Recent Developments. *Comb Chem High Throughput Screen.* 2009;12: 303–314. doi:10.2174/138620709787581666
3. Śledź P, Caflisch A. Protein structure-based drug design: from docking to molecular dynamics. *Curr Opin Struct Biol.* 2018;48: 93–102. doi:10.1016/j.sbi.2017.10.010
4. Blum A, Böttcher J, Heine A, Klebe G, Diederich WE. Structure-guided design of C2-symmetric HIV-1 protease inhibitors based on a pyrrolidine scaffold. *J Med Chem.* 2008;51: 2078–2087. doi:10.1021/jm701142s
5. Lindberg J, Pyring D, Löwgren S, Rosenquist Å, Zuccarello G, Kvarnström I, et al. Symmetric fluoro-substituted diol-based HIV protease inhibitors: Ortho-fluorinated and meta-fluorinated P1/P1'-benzyloxy side groups significantly improve the antiviral activity and preserve binding efficacy. *Eur J Biochem.* 2004;271: 4594–4602. doi:10.1111/j.1432-1033.2004.04431.x
6. Benach J, Swaminathan SS, Tamayo R, Handelman SK, Foltá-Stogniew E, Ramos JE, et al. The structural basis of cyclic diguanylate signal transduction by PilZ domains. *EMBO J.* 2007;26: 5153–5166. doi:10.1038/sj.emboj.7601918
7. Trott O, Olson AJ. AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *J Comput Chem.* 2010;31: 455–461. doi:10.1002/jcc.21334
8. Allen WJ, Rizzo RC. Implementation of the Hungarian algorithm to account for ligand symmetry and similarity in structure-based design. *J Chem Inf Model.* 2014;54: 518–29. doi:10.1021/ci400534h
9. Allen WJ, Balias TE, Mukherjee S, Brozell SR, Moustakas DT, Lang PT, et al. DOCK 6: Impact of new features and current docking performance. *J Comput Chem.* 2015;36: 1132–1156. doi:10.1002/jcc.23905
10. Kuhn HW. The Hungarian method for the assignment problem. *Nav Res Logist Q.* 1955;2: 83–97. doi:10.1002/nav.3800020109
11. Munkres J. Algorithms for the Assignment and Transportation Problems. *J Soc Ind Appl Math.* 1957;5: 32–38. doi:10.1137/0105003
12. Jones G, Willett P, Glen RC, Leach AR, Taylor R. Development and validation of a genetic algorithm for flexible docking. *J Mol Biol.* 1997;267: 727–748. doi:10.1006/jmbi.1996.0897
13. D.A. Case, I.Y. Ben-Shalom, S.R. Brozell, D.S. Cerutti, T.E. Cheatham, III, V.W.D. Cruzeiro TAD, R.E. Duke, D. Ghoreishi, M.K. Gilson, H. Gohlke, A.W. Goetz, D. Greene, R Harris, N. Homeyer YH, S. Izadi, A. Kovalenko, T. Kurtzman, T.S. Lee, S. LeGrand, P. Li, C. Lin, J. Liu, T. Luchko, R. Luo DJ, Mermelstein, K.M. Merz, Y. Miao,

- G. Monard, C. Nguyen, H. Nguyen, I. Omelyan, A. Onufriev, F. Pan R, Qi, D.R. Roe, A. Roitberg, C. Sagui, S. Schott-Verdugo, J. Shen, C.L. Simmerling, J. Smith RS, J. Swails, R.C. Walker, J. Wang, H. Wei, R.M. Wolf, X. Wu, L. Xiao DMY and PAK. AMBER 2018. 2018. Available: <http://ambermd.org/contributors.html>
14. Friesner RA, Banks JL, Murphy RB, Halgren TA, Klicic JJ, Mainz DT, et al. Glide: A New Approach for Rapid, Accurate Docking and Scoring. 1. Method and Assessment of Docking Accuracy. *J Med Chem*. 2004;47: 1739–1749. doi:10.1021/jm0306430
  15. Halgren TA, Murphy RB, Friesner RA, Beard HS, Frye LL, Pollard WT, et al. Glide: A New Approach for Rapid, Accurate Docking and Scoring. 2. Enrichment Factors in Database Screening. *J Med Chem*. 2004;47: 1750–1759. doi:10.1021/jm030644s
  16. O’Boyle NM, Banck M, James CA, Morley C, Vandermeersch T, Hutchison GR. Open Babel: An Open chemical toolbox. *J Cheminform*. 2011;3: 33. doi:10.1186/1758-2946-3-33
  17. Vento M, Cordella LP, Foggia P, Sansone C. A (sub) graph isomorphism algorithm for matching large graphs. *IEEE Trans Pattern Anal Mach Intell*. 2004;26: 1367–1372. Available: <http://amalfi.dis.unina.it/graph/>
  18. Hu J, Liu Z, Yu DJ, Zhang Y. LS-align: An atom-level, flexible ligand structural alignment algorithm for high-throughput virtual screening. Valencia A, editor. *Bioinformatics*. 2018;34: 2209–2218. doi:10.1093/bioinformatics/bty081
  19. RDKit: Open-source cheminformatics. Available: <http://www.rdkit.org>
  20. DockRMSD: Docking Pose Distance Calculation. Available: <https://zhanglab.ccmb.med.umich.edu/DockRMSD/>
  21. Desmet J, De Maeyer M, Hazes B, Lasters I. The dead-end elimination theorem and its use in protein side-chain positioning. *Nature*. 1992;356: 539–542. doi:10.1038/356539a0
  22. Dunbar JB, Smith RD, Yang CY, Ung PMU, Lexa KW, Khazanov NA, et al. CSAR benchmark exercise of 2010: Selection of the protein-ligand complexes. *J Chem Inf Model*. 2011;51: 2036–2046. doi:10.1021/ci200082t
  23. Pettersen EF, Goddard TD, Huang CC, Couch GS, Greenblatt DM, Meng EC, et al. UCSF Chimera--a visualization system for exploratory research and analysis. *J Comput Chem*. 2004;25: 1605–12. doi:10.1002/jcc.20084
  24. Wong DM, Greenblatt HM, Dvir H, Carlier PR, Han YF, Pang YP, et al. Acetylcholinesterase complexed with bivalent ligands related to Huperzine A: Experimental evidence for species-dependent protein-ligand complementarity. *J Am Chem Soc*. 2003;125: 363–373. doi:10.1021/ja021111w
  25. munkres - Munkres implementation for Python. Available: <http://software.clapper.org/munkres/index.html>
  26. Andersson HO, Fridborg K, Löwgren S, Alterman M, Mühlman A, Björnsne M, et al. Optimization of P1-P3 groups in symmetric and asymmetric HIV-1 protease inhibitors. *Eur J Biochem*. 2003;270: 1746–1758. doi:10.1046/j.1432-1033.2003.03533.x
  27. Biela I, Tidten-Luksch N, Immekus F, Glinca S, Nguyen TXP, Gerber HD, et al. Investigation of Specificity Determinants in Bacterial tRNA-Guanine Transglycosylase Reveals Queuine, the Substrate of Its Eucaryotic Counterpart, as Inhibitor. Randau L, editor. *PLoS One*. 2013;8: e64240. doi:10.1371/journal.pone.0064240