Protein design

CS/CME/BioE/Biophys/BMI 279

Oct. 29 and 31, 2024

Ron Dror

Final reminder: Please provide feedback to help us improve the course

- If you haven't already filled out the feedback survey, please do so immediately after class. It will close at 5 pm today.

- Anonymous survey:

https://forms.gle/rR6XBxeLp7Sy3m5k8

Watch these two recorded lectures before Thursday

- Fourier transforms and convolution

https://stanford-pilot.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=2f4f335a-76ac-401d-8577-b2140171fbfa

- Image analysis

https://stanford-pilot.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=9d9db1a4-b246-45af-8d54-b2140171fbec

Assignment 2

- Due Thursday Oct. 31 (1 pm)
- Those who wish to submit a solution to the challenge question may do so until Tuesday Nov. 5 (1 pm) but still need to submit the rest of the assignment by Thursday Oct. 31 (1 pm).

Outline

- What is protein design, and why do it?
- Sequence design
  - Traditional approach: energy minimization
  - Newer approach: machine learning
- Structure design
- Complementary experimental methods
- Large language models for protein design
- Examples of successful designs
- How well does protein design work?

What is protein design, and why do it?

Overall goal

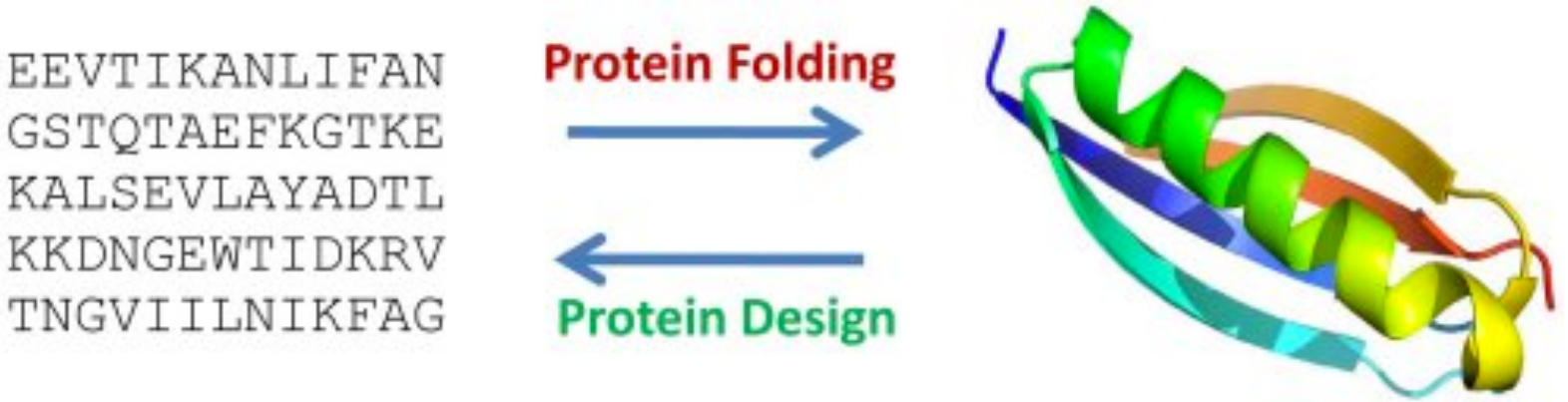
- Design a protein to serve a particular function or purpose
  - In particular, choose the appropriate amino acid sequence

Sample applications

- Designing enzymes (proteins that catalyze chemical reactions)
  - Useful for production of industrial chemicals and drugs
  - Potential environmental applications: degrading toxins or producing biofuels
- Designing proteins that bind specifically to other proteins
  - Potential for HIV, cancer, Alzheimer's treatment
  - Special case: antibody design
- Designing sensors (proteins that bind to and detect the presence of small molecules—e.g., by lighting up or changing color)
  - Calcium sensors used to detect neuronal activity in imaging studies
  - Proteins that detect TNT or other explosives, for mine detection
- Making a more stable variant of an existing protein (to facilitate experimental investigation)

Classical problem definition

- Given the desired three-dimensional structure of a protein, design an amino acid sequence that will assume that structure.
  - Of course, a precise set of atomic coordinates would determine the sequence. Usually we start with an *approximate* desired structure.
  - The problem of designing an amino acid sequence that will adopt a desired backbone structure is also known as “sequence design.”



http://www.riken.jp/zhangiru/images/sequence_protein.jpg

Note: the term “protein design” is sometimes used to describe several different (though related) problems

How do we choose the desired structure?

- Until recently, this was typically done with various *ad hoc* methods
- Lately, it's become more systematic, and is now sometimes called "structure design."
  - I will use that term here.

Note: the term "protein design" is sometimes used to describe several different (though related) problems

Typical protein design workflow

1. Based on design goals (e.g., desired function), choose structural requirements (e.g., some small part of the protein must adopt a particular structure, with the rest holding that part in place) — Human expert judgement
2. **Structure design**: select a backbone structure that is compatible with structural requirements and that is "designable" (i.e., "achievable" by real proteins) — Computation (or Human expert judgement)
3. **Sequence design**: select an amino acid that will adopt (fold into) the desired backbone structure — Computation
4. Perform wet-lab experiments to check which designs work, and (optionally) to improve them — Wet-lab experiments

Sequence design

Given the desired folded structure (specified by backbone coordinates), come up with a sequence that folds into that structure

The “direct” approach (doesn’t work in practice!)

- Given a target structure, search over all possible protein sequences
- For each protein sequence, predict its structure, and compare to the target structure
- Choose the best match

Why doesn't the "direct" approach work?

- Computationally intractable
  - Huge number of sequences to consider
    - 20^N possible sequences with N residues
- May not be good enough!
  - Protein structure prediction remains imperfect, especially for proteins substantially different from naturally occurring ones
  - We want to maximize the probability of the desired structure (compared to all other possible folded and unfolded structures)
    - We could do this by sampling the full Boltzmann distribution for each candidate sequence … but that's very difficult for even one sequence!

Sequence design

Traditional approach: Energy minimization

Simplify this problem by making a few assumptions

1. Assume the backbone geometry is fixed
2. Assume each amino acid can only take on a finite number of geometries (*rotamers*)
3. Assume that what we want to do is to maximize the energy drop from the completely unfolded state to the target geometry
   - In other words, simply ignore all the other possible folded structures that we want to avoid

The simplified problem

- At each position on the backbone, choose a rotamer (an amino acid type and a side-chain geometry) to minimize overall energy
  - Assume our energy function specifies a free energy. The Rosetta all-atom force field is a common choice.
  - For each amino acid sequence, energy is measured relative to the unfolded state.
  - Assume that energy can be expressed as a sum of terms that depend on one or two rotamers each. This is the case for the Rosetta force fields (and for most molecular mechanics force fields as well).
- Thus, we wish to minimize total energy E_T, where

$$E_T = \sum_i \left[E_i(r_i) + \sum_{i \neq j} E_{ij}(r_i, r_j) \right]$$

Note that r_i specifies both the amino acid residue at position i and that residue's side-chain geometry

Optimization methods

- Heuristic optimization methods are used in the great majority of protein design today
  - Not guaranteed to find optimal solution, but faster than exact methods
  - Most common is Metropolis Monte Carlo
    - Moves may be as simple as randomly choosing a position, then randomly choosing a new rotamer at that position
    - May decrease temperature over time (simulated annealing)

Optional: “Flexible backbone” design

- One of our key simplifying assumptions was that of a fixed backbone geometry.
- For many applications, protein design works better if you give the backbone some limited “wiggle room.”
- This requires optimizing simultaneously over rotamers and backbone geometry.
  - Often addressed through a Monte Carlo search procedure that alternates between local tweaks to backbone dihedrals and changes to side-chain rotamers
  - One can also refine a designed structure by local energy minimization, then re-optimize the side chains

Optional: Negative design

- Another simplifying assumption was that we simply minimize the energy of the desired structure
  - We do not consider all other possible structures. It's possible that their energy ends up even lower.
- In negative design, we identify a few structures that we want to *avoid*, and we try to keep their energies high during the design process.
  - This can help, but we cannot explicitly avoid all possible incorrect structures without making the problem much more complicated. So the overall approach is still heuristic.
- One can also perform structure prediction for designed sequences as a way to filter out ones that fold to an undesired structure

Sequence design

Newer approach: Machine learning

Lots of recent work on machine learning for protein sequence design

Protein sequence design with a learned potential

Namrata Anand[1], Raphael Eguchi[2], Irimpan I. Mathews[3], Carla P. Perez[4], Alexander Derry[5], Russ B. Altman[1,6] & Po-Ssu Huang[1]✉

Nature Communications, 2022

Generative models for graph-based protein design

NeurIPS, 2019

John Ingraham, Vikas K. Garg, Regina Barzilay, Tommi Jaakkola

Robust deep learning–based protein sequence design using ProteinMPNN

Science, 2022

J. Dauparas[1,2], I. Anishchenko[1,2], N. Bennett[1,2,3], H. Bai[1,2,4], R. J. Ragotte[1,2], L. F. Milles[1,2], B. I. M. Wicky[1,2], A. Courbet[1,2,4], R. J. de Haas[5], N. Bethel[1,2,4], P. J. Y. Leung[1,2,3], T. F. Huddy[1,2], S. Pellock[1,2], D. Tischer[1,2], F. Chan[1,2], B. Koepnick[1,2], H. Nguyen[1,2], A. Kang[1,2], B. Sankaran[6], A. K. Bera[1,2], N. P. King[1,2], D. Baker[1,2,4]*

And many more!

I will focus on the last of these (ProteinMPNN), as it's the first to be used widely for actual protein design

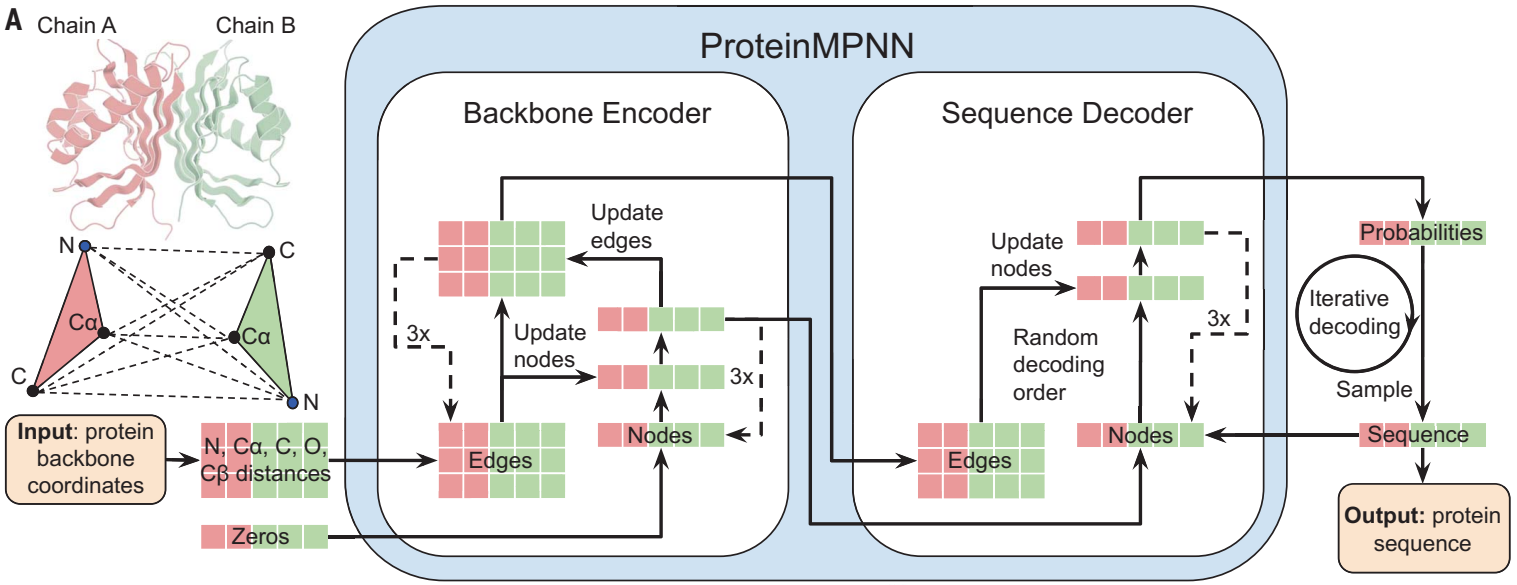
ProteinMPNN

- Basic idea: train a machine learning method to predict sequences of real proteins given their backbone structures
  - Importantly, add noise to the backbone coordinates in training
- ProteinMPNN goes through the sequence positions in a random order, predicting amino acid identities based on (1) backbone geometry and (2) amino acid identities already predicted at other positions
  - It can be run may times, generating a different sequence each time. Each of these is a candidate design.

Optional material

ProteinMPNN

- MPNN = message-passing neural network (operates on graphs)
- ProteinMPNN uses a graph to represent the backbone structure
  - That is, distances between atoms in the backbone



Dauparas et al., *Science* 2023

Structure design

What do we want our target structure backbone coordinates to be?

Designing the backbone

- The first step of protein design is generally to select one or more target backbone structures.
- Traditionally, this has been as much art as science
  - Apparently proteins can only adopt a limited set of backbone structures, but there wasn't a great description of what that set is.
- Traditional methods to design backbone structure:
  - Use an experimentally determined backbone structure
  - Use a fragment assembly program like Rosetta, selecting fragment combinations that fit some approximate desired structure
  - Assemble secondary structure elements by hand

Example of traditional backbone design

– To design "Top7," a protein with a novel fold, Kuhlman et al. started with a schematic, then used Rosetta fragment assembly to find 172 backbone models that fit it.

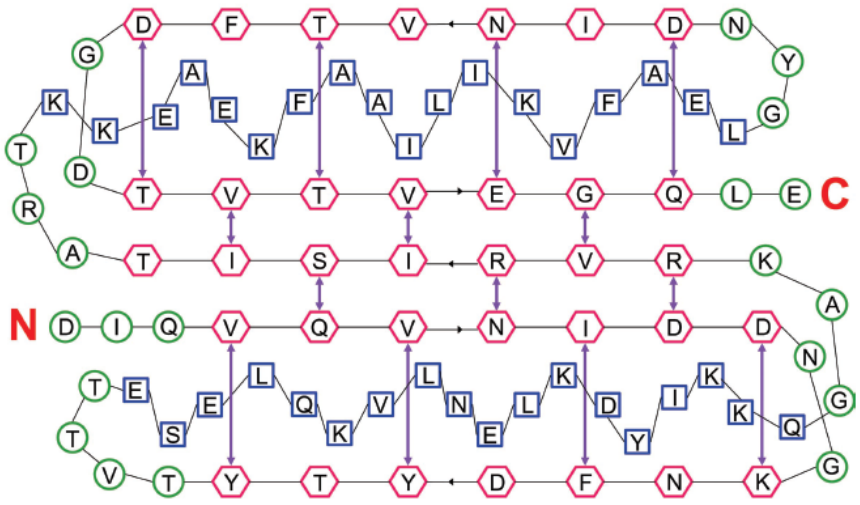
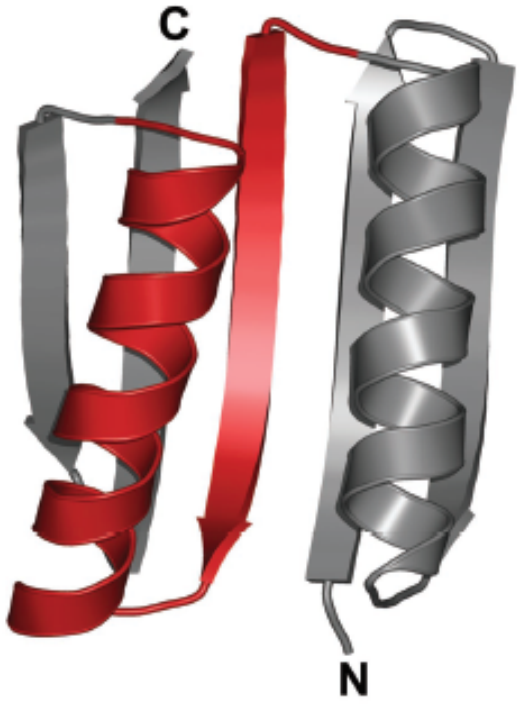


Fig. 1. A two-dimensional schematic of the target fold (hexagon, strand; square, helix; circle, other). Hydrogen bond partners are shown as purple arrows. The amino acids shown are those in the final designed (Top7) sequence.

Initial schematic of target fold. Hexagons = β sheet. Squares = α helix. Arrows = hydrogen bonds. Letters indicate amino acids in final designed sequence (these were not determined until much later).



Final structure

Kuhlman et al., Science 302:1364-8 (2003)

Could one use machine learning for backbone design?

- A challenging problem
  - This isn't simple prediction. Instead, it requires generating backbones that satisfy criteria for a given design and that will be adopted by one or more actual amino amino acid sequences
- A flurry of recent papers on this problem. I will focus on one particularly promising recent method.

Article Watson … Baker, *Nature*, 2023

De novo design of protein structure and function with RFdiffusion

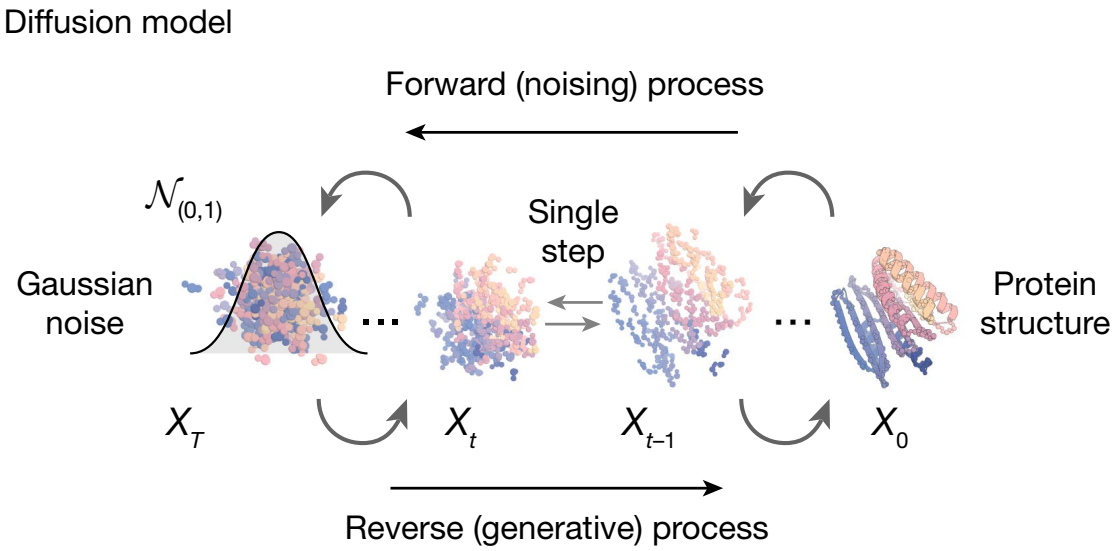
Structure design by RFdiffusion

- RFdiffusion (RoseTTAFold Diffusion) is based on the same machine learning approach as image generators like DALL-E: “denoising diffusion”
  - Originated in Surya Ganguli’s lab, Stanford Applied Physics Dept.

W.D. Heaven, “This avocado armchair could be the future of AI”, *Technology Review*, 2021

RFDiffusion

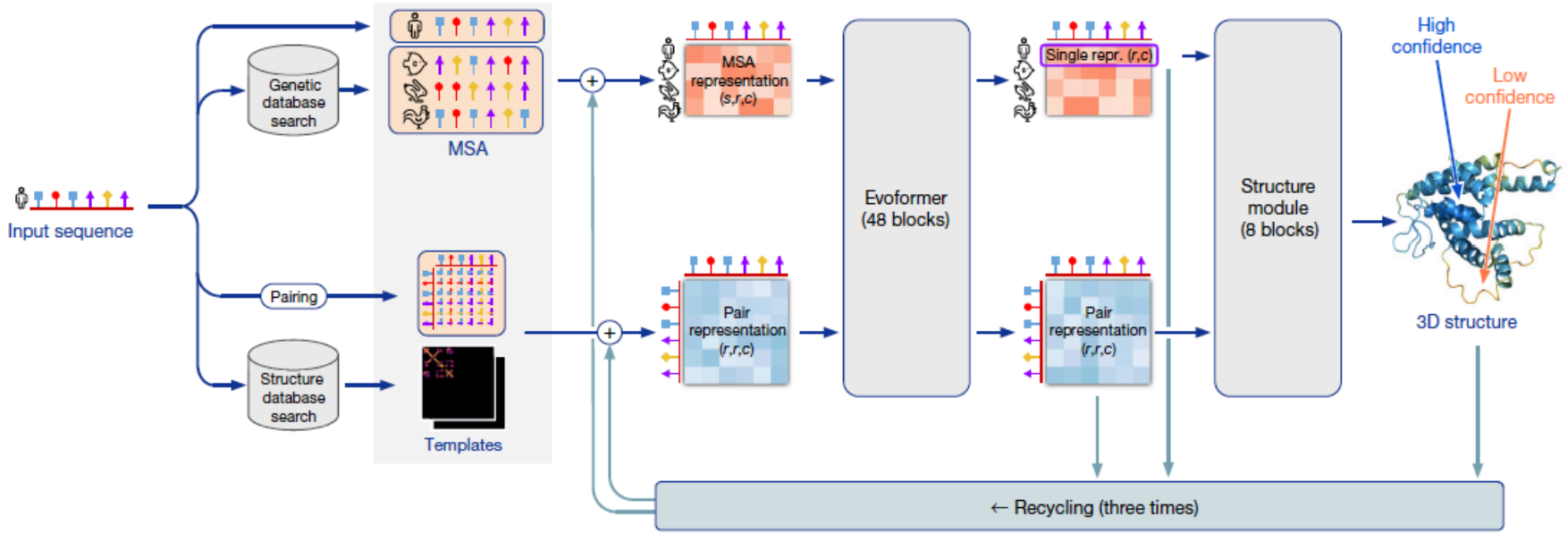
- Gradually convert protein backbone structures to random patterns by adding noise (i.e., random numbers) to the position and orientation of each amino acid
- Add this noise to the backbone atom coordinates a little bit at a time, over ~200 steps
- Train a machine learning method that, given the noisy ("messed up") backbone coordinates at one step, predicts the coordinates at the previous step (which are slightly less noisy)
  - This learned process is called the "denoising," "reverse," or "generative" process



Watson et al., *Nature*, 2023

RFDiffusion

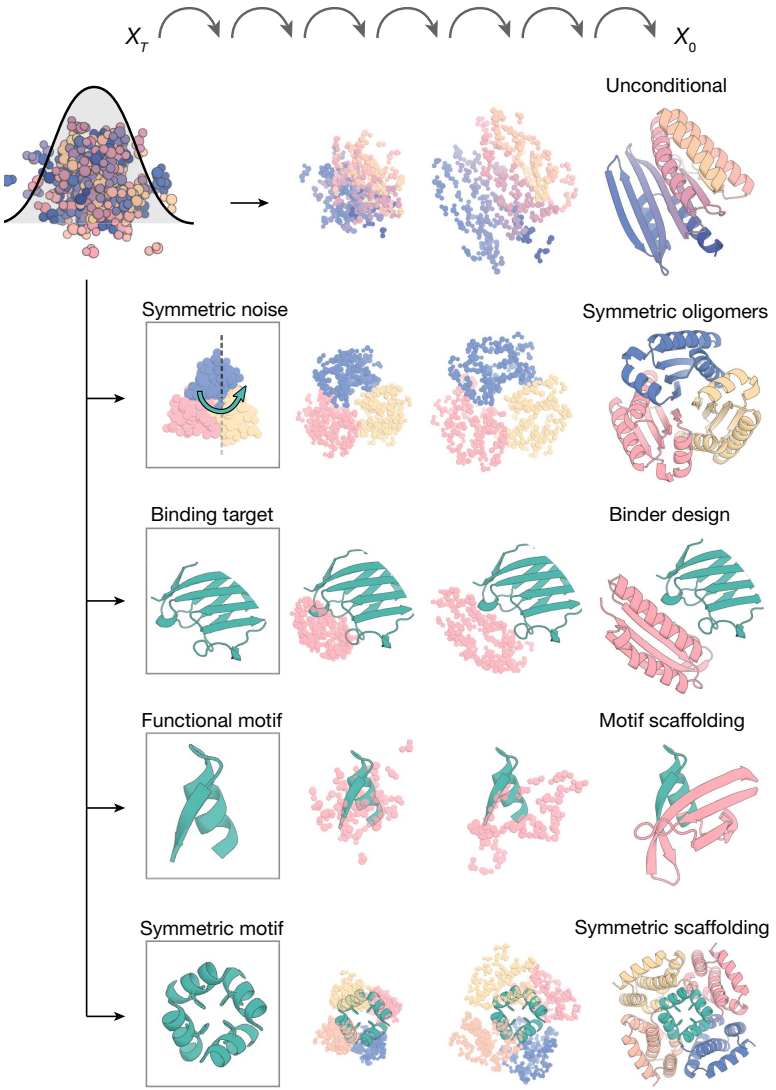
- Key insight: one can make this denoising process work much better by using a structure predictor as a component
  - RFDiffusion uses RoseTTAFold as a starting point, and tunes it for this application



AlphaFold 2 and recent versions of RoseTTAFold use “recycling,” meaning that they run the prediction process several times, feeding the output coordinates as inputs to the next prediction. This inspired the use of RoseTTAFold in RFdifussion.

RFDiffusion

- In making its predictions, the denoising (generative) process can use information on properties of the protein to be generated. This approach, known as “conditioning,” allows one to generate designs with desired properties.
  - For example, desired local structure (functional motifs), symmetry, or binding target



Watson et al., *Nature*, 2023

Complementary experimental methods

Computational protein design is often combined with experimental protein engineering methods

- For example, computational designs can often be improved by directed evolution
  - Directed evolution involves introducing random mutations to proteins and picking out the best ones
  - Usually this is done in living cells, with the fittest cells (i.e., those containing the “best” version of the protein) selected by some measure
- This is particularly powerful when designing proteins for a desired function that can be easily measured in cells

Frances Arnold

2018 Nobel Prize “for the directed evolution of enzymes”

Large language models for protein design

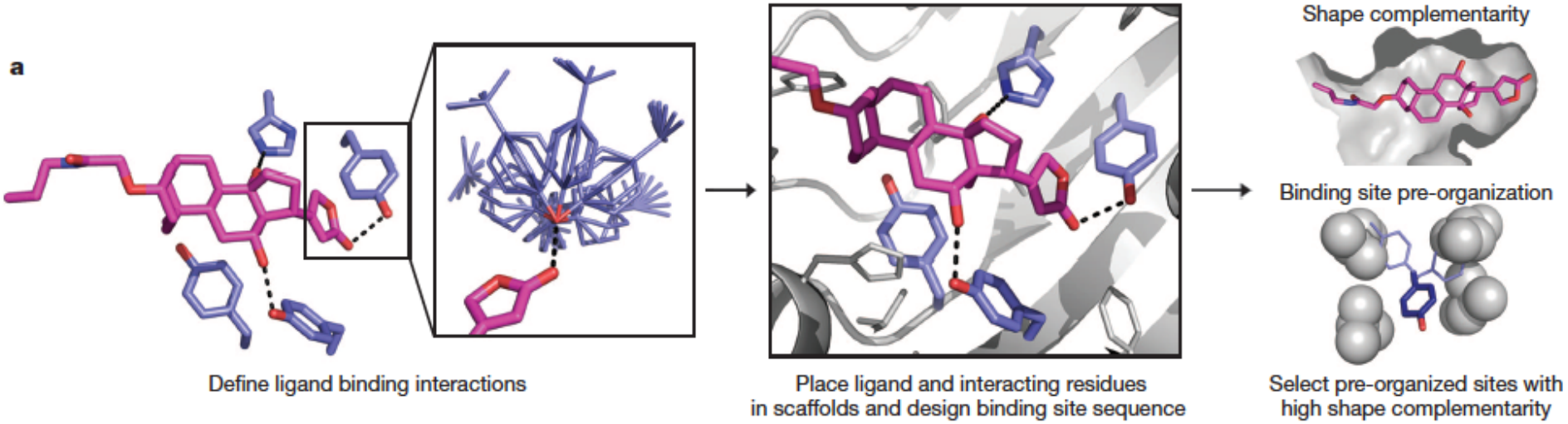
Large language models for protein design

- Protein large language models (e.g., the ESM models mentioned in the structure prediction lecture) have proven very useful for certain protein design tasks
  - These tasks include “optimizing” designed proteins to increase binding affinities or enzymatic activity
  - Nice example: Shanker … Hie, Kim, “Unsupervised evolution of protein and antibody complexes with a structure-informed language model”, *Science* 2024
- Some recent papers make much more general claims about structure and function design with language models alone. Some of these claims remain to be tested rigorously.

Examples of successful designs

Designing proteins that bind specific ligands

- The example below required specification of the position of certain side chains that will form favorable interactions with the ligand

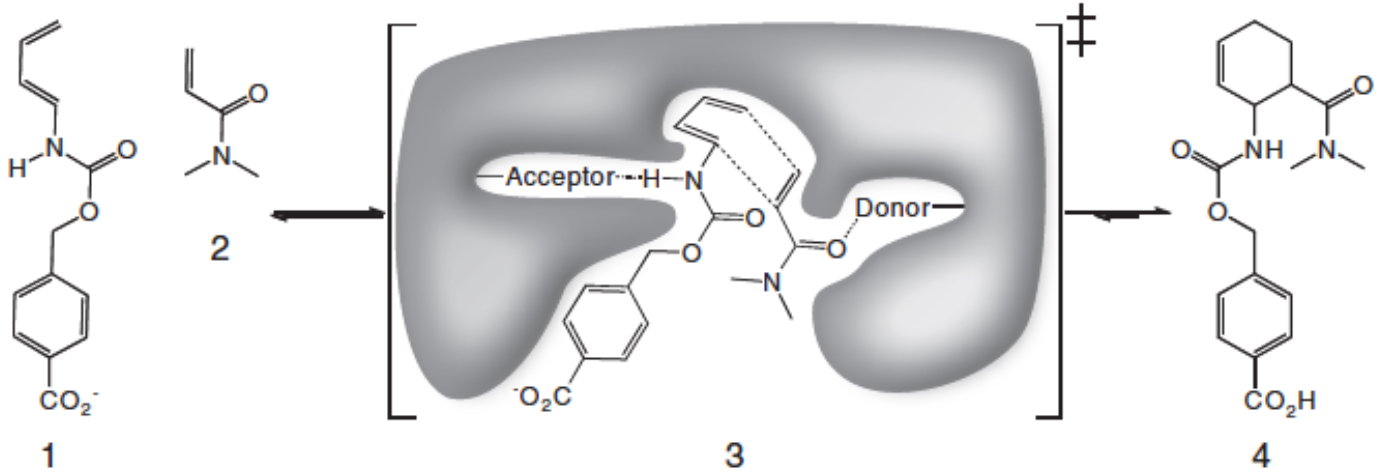


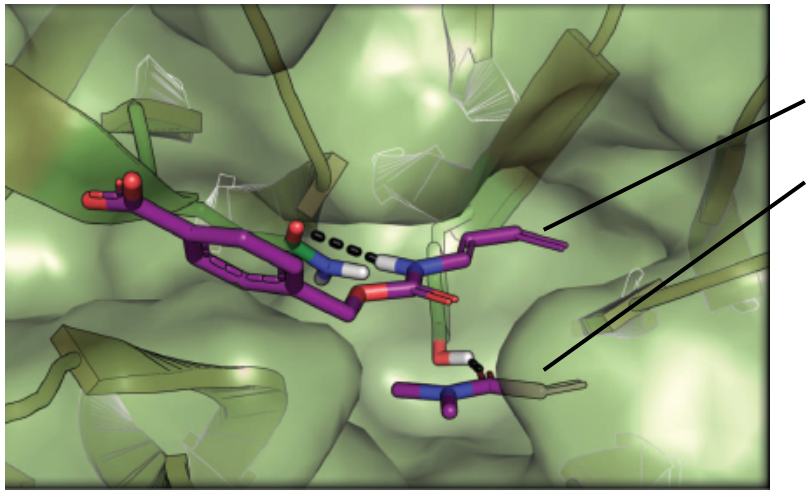
Protein designed to bind tightly to a specific steroid, but not to related molecules

Tinberg et al., Nature 501:212-6 (2013)

Designing enzymes

- In the example below, the protein holds two molecules in just the right relative positions for them to react. This speeds up the reaction. react more easily than when freely moving around



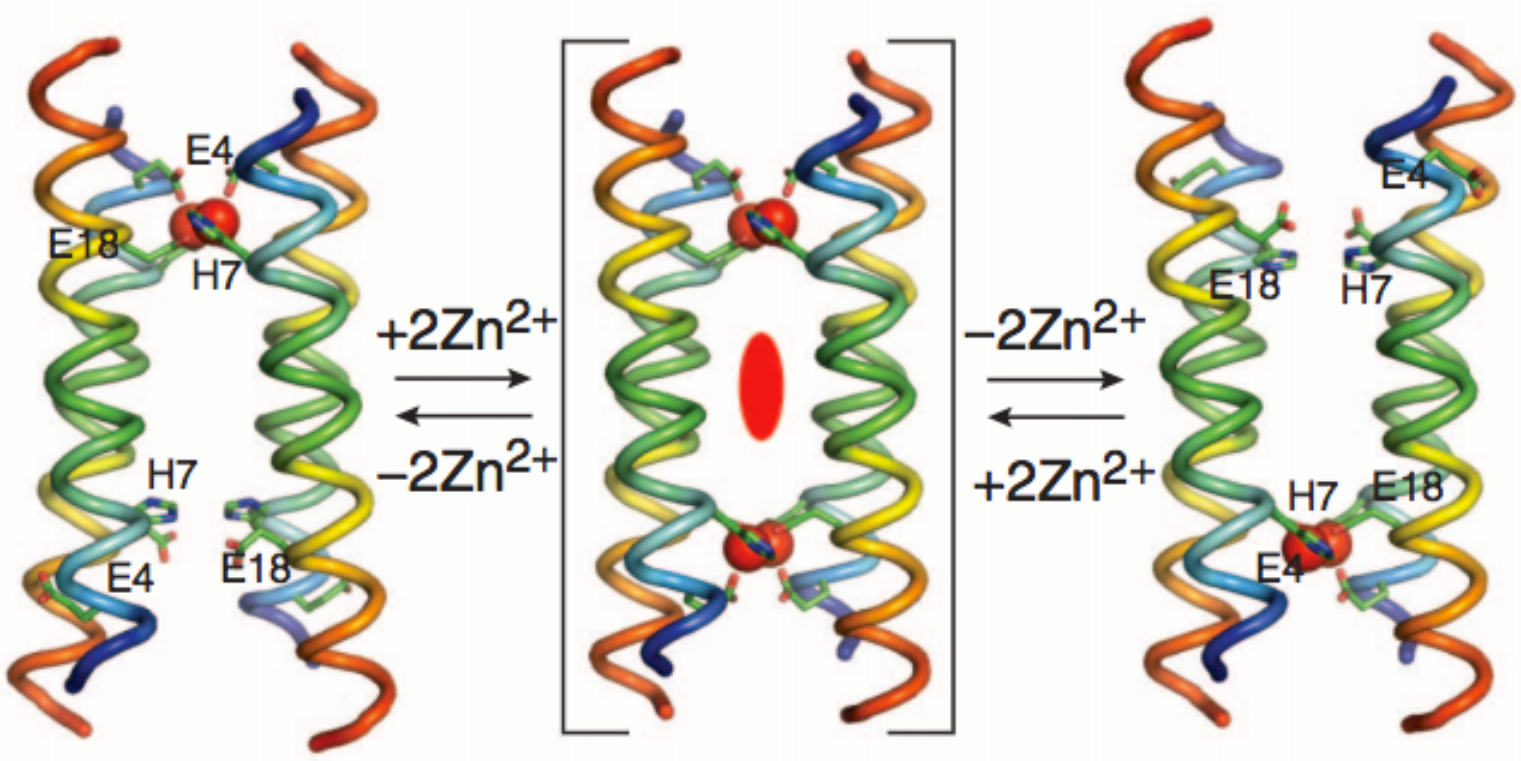


Molecule 1

Molecule 2

Siegel et al., Science 329:309-13 (2010)

Design of a transporter



- De novo design of a protein that transports zinc ions (Zn^{2+}), but not calcium ions (Ca^{2+}), across a cell membrane—a process that requires the protein to alternate between at least two conformations

transporter molecules are often extremely selective of one ion over others

Joh et al., Science 346:1520-24 (2014)

Designing multi-protein structures

Divine *et al.*, Designed proteins assemble antibodies into modular nanocages. *Science* 372:eabd9994 (2021)

"This week we report the design of new proteins that cluster antibodies into dense particles, rendering them more effective."

How well does protein design work?

How well does protein design work?

- Very impressive recent successes!
- However, one should keep in mind that:
  - Successful protein design projects often involve making and experimentally testing tens of candidate proteins (or more) to find a good one
  - Projects and design strategies that fail generally aren't published
  - Protein design is not yet a matter of simply "turning the crank," although machine learning methods like ProteinMPNN and RFDiffusion help automate it
- Evaluating/quantifying/comparing the effectiveness of protein design methodologies is difficult
  - Checking if a design "works" requires wet-lab experiments
  - To compare methodologies, one would need to synthesize and test many designed sequences for each methodology
  - One would need to do this for many protein design problems
  - Different protein design projects may have very different goals, so there isn't a universal metric for how "good" a given sequence is