Lecture 12: Protein structure prediction & design

BIO-212 – Biological Chemistry I

December 4, 2025

This week's topic will be on **Protein Design & Structure Prediction,** focusing on Google DeepMind's solution to the structure prediction problem: **AlphaFold**.

In the past, researchers looking to understand the structure of their favourite protein would usually have to solve its structure using one of the 3 methods learnt in last week's lecture: X-ray crystallography, Cryo-Electron Microscopy or NMR. This can sometimes be a tedious process, requiring many months or even years of experimental optimisation.

Then came in **Google DeepMind** with **AlphaFold 2** in 2021 which revolutionized structural biology. For the first time, a computational method could predict **accurate protein structures**—often approaching experimental quality—using only the amino-acid sequence. AlphaFold 2 learned patterns of evolution, co-variation, and geometry from large protein databases and could solve many previously unknown folds, completely transforming how scientists approach protein structure prediction. In addition to the structure, AlphaFold outputs **metrics** of its reliability that say something about the confidence one can have in its predictions.

Very recently (2024), another breakthrough emerged from Google DeepMind: **AlphaFold 3**. While AlphaFold 2 predicts only **protein** structures, AlphaFold 3 can model a much wider range of biomolecules, including **DNA, RNA, small molecules, metal ions, post translational modifications, glycans**, all within a single unified model.

We will be observing ourselves the power of AlphaFold 3 by using its online server: AlphaFold Server. For this all you need is an internet connection, a web browser and **a google account**. We will also be analysing some of the predictions using **PyMol** so make sure you bring a PC and ideally a mouse.

We will be predicting and analysing protein-DNA, protein-ligand, protein-ion and protein-protein structures. And as a final exercise, we will be *de novo* designing our own protein binder.

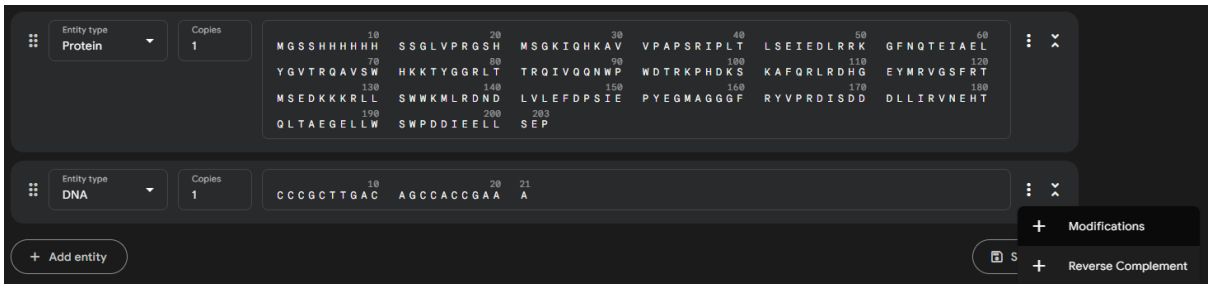
A couple of extra tips to make navigating PyMol easier:

- **If you ever get lost in your structure and want to "reset" your view, type into Pymol's command line:**
  orient
- **If you would like to zoom into a specific residue. In the sequence panel, right click the residue, and click on "zoom". Make sure that your selections are set to "Residues" and not "Chains"**

Exercise 1: 7r6r – Repressor protein:DNA Complex

For this first exercise we will analyse the structure of a mycobacteriophage immunity repressor in complex with its target DNA sequence. Go to the PDB deposit of **7r6r:** [RCSB PDB - 7R6R: Crystal Structure of a Mycobacteriophage Cluster A2 Immunity Repressor:DNA Complex](). Click on Display Files -> **FASTA sequence**

This should open a tab displaying the amino acid sequence of the repressor protein, as well as the DNA sequence that it is bound to. Copy paste the amino acid sequence of **Chain A.** Go to the AlphaFold 3 Server: AlphaFold Server, delete the default sequences that are already there, then click on **Add entity.** Make sure that the **Entity type** is set to Protein, then paste the sequence into the Input. Then, go back to the FASTA sequence of **7r6r** and do the same for **Chain C** (DNA sequence) but for this make sure that the **Entity type** is set to DNA. Then in the options on the far right, add the **Reverse Complement** DNA sequence:



This will automatically add the complementary, antiparallel DNA sequence as a sequence to the prediction.

Click on **Continue and preview job,** keep the Seed set to **auto**, name the job something easy to recognise, then **Confirm and submit job** and wait for the prediction to be completed. **Each AlphaFold 3 prediction done during this week's exercise session will only take max 5 minutes.**

1. Once the prediction is completed, click on it and briefly analyse the structure in the 3D-viewer. AlphaFold 3 Server colours the output based on its per-atom **plDDT** metrics.
    a. Explain briefly what the plDDT metric is.
    b. Take a screenshot of the prediction. Based on what you see in the 3D-viewer, evaluate the plDDT of this prediction. Is AlphaFold 3 confident about the predicted model?
2. Another metric that AlphaFold outputs is the PAE and pTM. The PAE can be seen as a matrix on the right side of the screen. The iPTM and pTM are written above the 3D-viewer.
    a. Briefly describe what the PAE and pTM metrics tell us. Evaluate these metrics for the prediction. Take a screenshot of the PAE matrix. Which part of the protein has the worst PAE metrics?

3. Download the files (top of the screen) and extract the zip files. But before we look at the predictions on PyMol, let's first analyse the experimentally determined structure. Download and open **7r6r.pse** from Moodle.
    a. Given the highly hydrophilic phosphodiester backbone in DNA, DNA binding proteins tend to have highly polar interfaces with DNA duplexes. In this PyMol session, all residues in the repressor protein within **4 angstroms** from the DNA duplex have been selected, coloured in cyan and shown as sticks. The interactions within the protein-DNA complex along with the interactions within the DNA duplex itself have been shown. Find 2 examples (show screenshots) of polar interactions (hydrogen bonds or ionic bonds) between the bacteriophage repressor protein and its target sequence. Describe the type of interaction in each case.
    b. In AlphaFold 3's output files, you should find **5 predicted models** of the repressor-DNA complex. Upload **"…_model_0"** into the **7r6r** PyMol session by dragging and dropping the model_0 file directly into the PyMol session. Alternatively in PyMol, go to File -> Open… then select and upload the file. In order to quickly evaluate how close AlphaFold 3 was to the experimental structure, type into PyMol's command line **(make sure to unclick sele_polar_conts and that only 7r6r and the AlphaFold prediction are selected)**:

    alignto

    Take a screenshot of the two aligned structures. What is the RMSD of the AlphaFold 3 model to the original structure? How good is the prediction, in your opinion? What are some differences that you observe?

Exercise 2: 8cvp – Cereblon:DDB1

Now that we've seen the power of AlphaFold 3 in predicting accurate structures of biomolecules, let's see a case where it may get the prediction wrong. Take the case of the **Cereblon:DDB1 complex**. Cereblon (CRBN) is a substrate receptor of the CRL4 E3 ubiquitin ligase, and it binds to the adaptor protein DDB1 to form the core of the ubiquitination machinery.

Together, the CRBN–DDB1 complex recruits specific protein substrates for ubiquitination, and it is the pharmacological target of drugs like thalidomide and lenalidomide, which bind Cereblon and redirect the complex toward novel degradation targets.

The structure of the CRBN:DDB1 complex was solved both in the presence (PDB ID: **8d7u**) and absence (PDB ID: **8cvp**) of a drug. When a drug is present, Cereblon binds it and then undergoes a **conformational change.**

1. Open the **CRBN_DDB1.pse** PyMol session from Moodle. Here you will the CRBN:DDB1 complex solved both in the **presence** and **absence** of a drug. Identify which one of the two chains is Cereblon, and where the drug binds. Can you see the conformational change?

Now let's see if AlphaFold 3 can predict the correct structure. Go to the AlphaFold 3 server. To save compute time, we will **only** predict the structure of **Cereblon**, with the Zinc ion. Clear the input from the last prediction, and click on **Add entity**, and add the following sequence:

MEEFHGRTLHDDDSCQVIPVLPQVMMILIPGQTLPLQLFHPQEVSMVRNLIQKDRTFAVLAYSNVQER EAQFGTTAEIYAYREEQDFGIEIVKVKAIGRQRFKVLELRTQSDGIQQAKVQILPECVLPSTMSAVQLESL NKCQIFPSKPVSREDQCSYKWWQKYQKRKFHCANLTSWPRWLYSLYDAETLMDRIKKQLREWDENL KDDSLPSNPIDFSYRVAACLPIDDVLRIQLLKIGSAIQRLRCELDIMNKCTSLCCKQCQETEITTKNEIFSLS LCGPMAAYVNPHGYVHETLTVYKACNLNLIGRPSTEHSWFPGYAWTVAQCKICASHIGWKFTATKKD MSPQKFWGLTRSALLPTIPD

Then click on **Add entity,** and for the Entity type click on **Ion** and select **Zn^{2+}**. Click on **Continue and preview job** and submit the job.

2. Once the prediction is done, open the results on the server. What does the PAE plot look like this time? Analyse the plDDT of the structure. Which parts of the structure is AlphaFold not so confident about? Why?
3. Download the zip file results and extract the file. Drag and drop the **"…model_0"** prediction into the PyMol session. Align the prediction first with **8cvp**, then with **8d7u**. **(*If the structure seems to disappear after alignment, type in "orient" into the Command line!)***
    a. What are the RMSDs after aligning the prediction to both structures?

b. Therefore, which state does the AlphaFold 3 prediction resemble more, the **apo** (ligand unbound) or the **holo** (ligand bound) state? Why?

Exercise 3: Ran GTPase (again)

For the third exercise, we're going to see how well AlphaFold 3 can predict the structure of our favourite GTPase: Ran.

Last week we saw how binding GDP or GTP (or GNP, a GTP analogue) can change the conformation of an important GTPase involved in nuclear transport. Now let's see if AlphaFold can predict these conformational changes.

Go to the AlphaFold 3 server again and input the following protein sequence:

EPQVQFKLVLVGDGGTGKTTFVKRHLTGEFEKKYVPTLGVEVHPLVFHTNRGPIKFNVWDTAGQEKFG GLRDGYYIQAQCAIIMFDVTSRVTYKNVPNWHRDLVRVCENIPIVLCGNKVDIKDRKVKAKSIVFHRK KNLQYYDISAKSNYNFEKPFLWLARKLIGDPNLEFVAMPALAPPEVVMDPALAAQYEHDLEVAQTT

Then add another entity, and for the Entity type, click on **Ligand**, then amongst the options, click on **GDP, Guanosine-5'-diphosphate.**

Then add another entity, and this time add a Mg^{2+} ion.

Submit the job, and don't forget to name the job something easily recognisable.

1. Once the job is completed, analyse the metrics (PAE, plDDT, pTM). How confident is AlphaFold about this prediction?
2. Download and extract the files. Download and open **Ran.pse** on Moodle. Drop the **"…model_0"** prediction into the PyMol session. Align the prediction to **1byu**. What is the RMSD? Does AlphaFold 3 do a good job in predicting Ran's overall structure? What about the GDP binding pocket, and the Mg^{2+} placement?

Now go back to the AlphaFold 3 server, and submit the same input, but this time change the Ligand Entity from **GDP to GTP.** Submit the job, and name the job something easy to differentiate it from the previous prediction.

3. Analyse the metrics, especially the plDDT and PAE metrics. Is AlphaFold 3 as confident about Ran's structure in the GTP bound state? Which parts of the protein are predicted with less confidence?
4. Download and extract the files. Align the prediction with **1byu** and then with **1rrp**. Which one has the lower RMSD? Does AlphaFold 3 capture the "active" GTP bound state well?

Exercise 4: 2v3m – NAF1 Homodimer

2v3m is the crystal structure of the Naf1 homodimer, a protein involved in RNA processing and telomerase biogenesis. Since another incredible feature of AlphaFold is its capacity to model protein complexes, let's see if AlphaFold 3 can recapitulate the correct dimeric assembly.

Go to the AlphaFold 3 server and input the following NAF1 sequence:

ETVPELPEDYEISEKTIITPIGVLKSAFENNIIIHATMSGEKRVLKEGSIFCLEDRTLIGMLTEVFGPLQNPF YRIKLPDSKKNLFDELKVRLGEKAFIVT

Then add another Protein Entity, and input the same sequence again. Submit the job.

1. Another metric that AlphaFold 3 outputs is the ipTM. Briefly describe what this metric tells us. Analyse the ipTM metric, is AlphaFold 3 confident about the homodimer prediction?
2. Download the files, and load the **"…model_0"** prediction unto the PyMol session **2v3m.pse** from Moodle. Align the prediction with **2v3m. (*If the structure seems to disappear after alignment, type in "orient" into the Command line!)*** What is the RMSD? Does AlphaFold 3 do well in predicting the NAF1 homodimer?

Exercise 5 (Optional structure prediction exercise): 7bbv – Pectate Lyase

Next, we will explore the structure of a **pectate lyase (plant degrading enzyme)** from the fungi *Verticillium dahliae*. The **7bbv** structure shows this enzyme decorated with mannose sugar modifications on its surface. Additionally, the enzyme is stabilised by a calcium ion, which binds in a specific pocket. This is a perfect example of a structure AlphaFold 3 was trained to predict.

Go to the AlphaFold 3 Server, and this time input the following amino acid sequence:

ASVTESCNIGYASTNGGTTGGKGGATTTVSTLAQFTKAAESSGKLNIVVKGKISGGAKVRVQSDKTIIG QKGSELVGTGLYINKVKNVIVRNMKISKVKDSNGDAIGIQASKNVWVDHCDLSSDLKSGKDYYDGLL DITHGSDWVTVSNTFLHDHFKASLIGHTDSNAKEDKGKLHVTYANNYWYNVNSRNPSVRFGTVHIY NNYYLEVGSSAVNTRMGAQVRVESTVFDKSTKNGIISVDSKEKGYATVGDISWGSSTNTAPKGTLGSS NIPYSYNLYGKNNVKARVYGTAGQTLGF

On the far right, click on options (3 dots) and then add **PTMs** (Post-Translational Modifications). In the sequence, click on **Threonine (T) 4**, then add a **Glycan chain**. For the sequence of glycans to add, type in **MAN**. This lets the software know that threonine 4 should be modified by a single Alpha-D mannose residue. Do the same for the following 5 amino acids:

Threonine (T) 26

Threonine (T) 27

Threonine (T) 28

Serine (S) 30

Threonine (T) 36

Then, click on **Add entity**, change the **Entity type** to **Ion**, then choose Ca^{2+}. Submit the job.

Once the prediction is completed, download the zip file and extract the contents. Download and open the **7bbv.pse** from Moodle. Upload the **"…model_0"** prediction into the PyMol session as done before. Align the two structures.

1. Take a screenshot of the two aligned structures. What RMSD did you get after alignment?
2. How well does AlphaFold 3 predict the binding site of the calcium ion?
3. How well does AlphaFold 3 predict the mannose modifications?
4. Let's take a deeper look into the predicted calcium binding site. Unclick the **7bbv** object in the object panel, so that only the predicted model is in view. Zoom in into the calcium binding site – Which 3 residues are stabilising the calcium ion in the pocket?

Design your own *de novo* binder!

While tools like AlphaFold deal with the folding problem, many others try to solve the so-called **"inverse folding"** problem, that is: given a protein structure, which are the amino acid sequences that will fold into that shape? This task is also known as protein design and it can used to engineer existing proteins to enhance their qualities (solubility, thermostability, activity...) or to design ***de novo*** proteins for any purpose. One such purpose that we will explore in this workshop is the design of protein binders.

Open in your browser the link to the ColabDesign notebook: diffusion.ipynb - Colab

This notebook conveniently joins three very useful tools for protein design: **RFDiffusion**, **ProteinMPNN**, and **AlphaFold (AlphaFold 2 here)**. RFDiffusion can generate protein **backbones** of a certain length with or without guidance from the user. ProteinMPNN can then assign several different **amino acid sequences** that should fold into the protein backbone generated by RFDiffusion. Finally, we can use AlphaFold to check if the sequence generated by ProteinMPNN is predicted to fold into the shape originally generated by RFdiffusion. This way, we have a sturdy and iterative in silico pipeline that gives us more or less confidence about whether a designed protein sequence will fold as intended or not. Now, we'll use this pipeline to generate a peptide binder to the interleukin-7 receptor alpha PDB ID: **3di3**

Prepare the following settings:

For the RFdiffusion block:

- Name: binder_design_1
- contigs: 20-100/0 B17-209
    - *This tells RFDiffusion to generate a protein of length 20-100 amino acids as a single chain ("/0") to bind residues 17-209 of chain B (the interleukin 7 receptor)*
- pdb: 3DI3
- num_designs: 1

None of the other RFdiffusion settings need to be changed

Then for the ProteinMPNN & AlphaFold block:

- num_seqs: 8
- initial_guess: ticked
    - *This tells AlphaFold to use the crystal structure of the receptor as a template for its prediction in complex with the binder.*
- num_recycles: 3
    - *The more recycles the better! (…and the longer the prediction takes)*
- use_multimer: ticked

Leave the remaining settings as is. Once you're done go to **Runtime** (at the top of your screen), then **Run all.** The run should take approximately 7-10 minutes. Make sure your laptop doesn't go to sleep in this time.

Once the binder generation process is completed, a zip file containing the output should be automatically downloaded. Extract the files from the zip file. Download and open **3di3.pse** from Moodle. You can find the best designed binder from the **outputs-> binder_design_1 -> best_design** file downloaded from the notebook. Drag and drop this file into the **3di3** PyMol session. Align the structures.

1. Based on the alignment, do you think your binder will compete with the native interleukin-7 for its receptor?
2. We can also evaluate the plDDT of the prediction on PyMol. Unclick **3di3**. Then for the **best_design,** colour -> spectrum -> b-factors. AlphaFold keeps its plDDT information in what is known as the **b-factors** of the pdb file. PyMol can read this information, then colour each residue on a red (high plDDT) to blue (low plDDT) colour scale. Based on the colours you observe, is AlphaFold confident about the structure of the binder?
3. If you're interested in seeing what RFdiffusion's diffusion trajectories look like, go to the outputs file -> traj and drop the file ending in **"…Xt-1_traj"** into the PyMol session. On the bottom right, there should be a play button. This allows you to visualise the denoising trajectories RFDiffusion takes to generate the mini binder!