YouSet Superimposition Builder

Table of contents

Objective

The main objective of this project is developing a method to model the macro-complex structure of biomolecules, formed by proteins and nucleic acids (DNA or RNA) with information of each pairing interaction of a complex. The program must return a PDB file with the best macro-complex approximation.

Input files

The input files consists in pairs of interacting chains in PDB format, that can be obtained with a module included in our program, designed exclusively to obtain these interactions from the original macro-complex in PDB format.

Program design

Obtaining input PDB files

Find the chains interactions from PDB files it is not part of our program but we have developed a module to split macro-complexes in pdb files of pairs of all type of chains interacting (protein-protein, protein-nucleic acid and nucleic acid-nucleic acid). This module is called interaction.py. We considered that two protein chains interact if the distance between alpha carbons (CA) is less than 8 Å, at least once. Between protein-nucleic acid we considered interaction if at least one CA of the protein chain is below 8 Å of any atom of the nucleic acid chain. Finally, nucleic acid-nucleic acid interactions have been considered if at least one distance is below 8 Å between any of the atoms of both chains.

Modeling the macro-complex

Read input files and store interactions

The program is able to extract information of interactions between pairs of chains that are stored in individual pdb files. The input should be the path to the directory where these pdb files are stored. Only pdb files with the following format are accepted:

- 4 initial characters corresponding to the name of the complex.
- Followed by 2 characters that represent the ID of the two chains that are stored in that pdb file.
- Followed by the .pdb extension.

For example, the files “5nssAB.pdb” and “3t72c1.pdb” would be accepted, but not “5nss_AB.pdb” or “3t72c1B.pdb”. Interactions from PDB files are stored in a dictionary, with the chain identifiers as keys, and a list containing the interactions in which that particular chain participates.

Chain classification

The module sequencesalingment.py is able to read the multifasta file. The header of each sequence must be in the following format:

- The character “>”
- 4 characters corresponding to the name of the complex.
- The character “:”
- 1 character representing the ID of the chain

  For example: 4GH3:A If a header of the fasta file does not present this format, it raises an exception indicating which sequence has the problem and which should be the correct format. The ID given for each sequence must correspond to the ID given in the pdb file.

Then, it checks if the sequence corresponds to nucleic acids or proteins and it performs pairwise sequence alignments to check for similar sequences. Two chains will be considered the same if they share more than 95% of identity, although they have different IDs. This will allow us to work with the real stoichiometry of the complex, depending on the sequence and not on the chains IDs.

Stoichiometry reading

The user is free to select the stoichiometry of each chain from the command-line, which means defining the number of subunits of each chain type included in the macro-complex. If the stoichiometry of the complex is not defined, by default the stoichiometry is established as follows: each type of chain will have as many subunits as different identifiers for that chain have been given by the user.

Macro-complex building

The macro-complex will be obtained using a recursive function based on superimposition, joining pairs of interacting molecules in a single PDB file. The program will run for complexes of only proteins, and complexes of proteins and nucleic acids. To obtain the macro-complex, we designed a module called complexbuilder.py to superimpose chains according to their interactions.

This module starts by taking a first interaction and establishing it as the initial complex. Then it iterates for the rest of the dictionary, trying to add new chains to this initial complex. To do this, it takes another pair of interacting chains and checks if there is at least one identical chain in the macro-complex in order to do the superimposition. If not, it stores this interaction for the next iteration of the function.

It also uses the information of which chains have been already included in the macro-complex and how many chains of each type can still include (stoichiometry). In case it cannot add any chain of a given interaction, it discards the whole interaction. If only one of the chains can be included, it selects that one, from now on referred as "new chain", as the one to be added to the complex. The other chain, the one that is already in the complex, will be called "common chain". In case that both chains can be included, it select the first one as the "common chain" and the second one as the "new chain".
Then, it tries to add the movil chain to the complex. This is done by fixing the common chain and superimposing it one by one to all identical chains that are already in the complex, recursively. Then, it uses the rotation and translation matrices to transform the atomic coordinates of the new chain. At this point, the program is able to detect clashes between the chains of the complex and the new chain. We identified as clashes:

- **Protein - protein** interactions with at least one distance between CA atoms less than 2 Å.
- **Protein - nucleic acid** interactions with at least one distance between any atom of the protein chain and any atom of the nucleic acid chain less than 1 Å.
- **Nucleic acid - nucleic acid** interactions with at least one distance is below 1 Å between any of the atoms of both chains.

We have decided these distances because we have found that CA steric clashes (Alpha-carbon atom pairs) have higher cut-off distances, and steric clashes (any atom pair) nearly 1 Å.
If a clash is detected, the program will automatically stop to try to add the new chain in that position and will continue iterating by all the possible positions. If finally this chain cannot be included, the whole interaction will be stored for the next iteration of the function.
This function will iterate a maximum of 10 times in order to try to add all the possible chains in the complex, according to the given stoichiometry. When finished, it stores the complex model in a pdb file. If some interactions have not been added to the complex, it prints a message in the standard error, specifying the identifiers of the interaction.
Moreover, if the option verbose had been previously selected, the program gives the final stoichiometry of the model, considering protein chains and nucleic acids separately.

Organization of the package

This package is composed by the following modules:

- buildcomplex_clear.py: is the main module that contains the workflow of the program.
- sequencesalignment.py: it performs pairwise sequence alignment to detect those chains that we will consider equals.
- read_stechiometry.py: it reads the stoichiometry given by the user if the option stoichiometry is selected.
- complexbuilder.py: it contains several functions that allow the formation of the complex.

Tutorial

Installation

Via pip

Go to the terminal and install with the following command:

```
pip install YouSet
```

Then locate the installation directory of pip to use the main script *buildcomplex_clear.py*:

```
pip show YouSet
```

This option also installs the required biopython package. Numpy is installed as it is required for some functions of biopython.

Via GitHub

Clone the GitHub repository containing the source code by using the following command:

```
git clone https://github.com/gsergom/YouSet.git
```

From source code

Download the source code of the current release and untar the files:

```
wget https://github.com/gsergom/YouSet/archive/1.6.tar.gz
tar -zxvf 1.6.tar.gz
```

> Note: Please replace the "1.5" for the desired version of the program on the command line!

Obtaining input files

To obtain the input files corresponding to pdb files of each pair of chains interacting, you can include the module interaction.py inside the same directory where you have included the macro-complex pdb file. Once you have these two files in the same directory, execute the following command, where the last argument is the name of the complex without the .pdb extension: python3 interaction.py 6gmh

Building the macro-complex

To obtain the macro-complex in a single pdb file, the user has to execute the script buildcomplex_clear.py from the command line. The user can specify several arguments, that can be optional or mandatory, as the directory with all of the interacting pairs of chains, the name of the output directory, etc. These different options are accessible with the following commands:

```
python3 buildcomplex_clear.py -h
python3 buildcomplex_clear.py --help
```

Options:

- Optional arguments:
  -h, --help: show a help message and exit
  -v, --verbose: print log to stderr
  -s --stoichiometry: Optional stoichiometry of the macrocomplex
- Mandatory arguments:
  -d INDIR, --pdbdir INDIR: path to directory with all the base pdb files to use.
  -f INFASTA, --fasta INFASTA: path to fasta file with the sequences of the complex.
  -o OUTDIR, --output OUTDIR: path to output directory where to store the results.

To run the program with the defaults settings, the following command must be used:

```
python3 buildcomplex_clear.py -d 5nss_interact/ -f 5nss.fasta -o 5nss_interact/results/
```

If the user want obtain the final stoichiometry of the model, the option verbose should be used. When the final model is build, it will print the following message:

```
The stoichiometry of the final model is A2C1D1M1E1F6 + 2 nucleic acid(s): H1I1.
```

If the user want to define the stoichiometry, the option stoichiometry should be used. In that case, the program will ask the user to introduce in the command line, one chain at a time, the number of times that that chain must appear in the complex, indicating the equivalences that have been found by the sequences alignment. Only positive integers are considered as valid values. Otherwise, the program will print the corresponding error message until a valid value is entered.

```
8 different chains were found.
Provide the number of times that each chain appears in the complex and press ENTER.
Chain A (these chains are considered the same: A,B): 4
Chain C (these chains are considered the same: C): 4
Chain D (these chains are considered the same: D): 4
Chain E (these chains are considered the same: E): 4
Chain F (these chains are considered the same: F, G, J, K, L, N): 4
```

Theoretical background

The interactions in a macro-complex define how chains are displayed in space, which chains surround other chains, and how they interact with each other through forces and contacts. We considered that two chains interact with distances between atoms less than 8 Å. For protein-protein interactions at least one distance between CA atoms, for protein - nucleic acid at least one distance between CA atoms of the protein chain and all atoms of the nucleic acid, and between nucleic acid chains at least one distance between all atoms was considered.

Clashes are unfavorable interactions where atoms are too close in space and can have a problem of energetic repulsion. Protein-protein clashes have been considered as interactions with at least one distance between CA atoms less than 2 Å. We have found that CA steric clashes (Alpha-carbon atom pairs) have higher cut-off distances than steric clashes (any atom pair). Protein-nucleic acid and nucleic acid-nucleic acid interactions, have been considered steric clashes if at least one distance is less than 1 Å between any atom. This program have been designed to avoid clashes in the final model, if one clash is detected, it stops to try to add the new chain in that position and continues iterating by all the possible positions.

The program will iterate a maximum number of times in order to try to add all the possible chains in the complex, according to the given stoichiometry. This stoichiometry may be given by the user with an optional argument, but if it is not selected, the program holds by default the real stoichiometry of the complex.

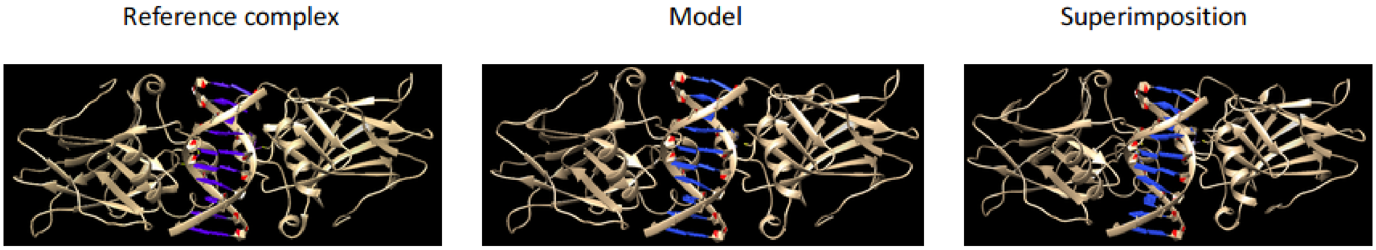
Once the models of the examples have been obtained, we compared both original and model complexes with Chimera, using the superimposition function MatchMaker. It superimposes protein or nucleic acid pdb structures by creating pairwise sequence alignments and fitting the aligned residue pairs. We can partially know if the model is a satisfactory macro-complex approximation with two ways: checking for differences in the MatchMaker superimposition, and checking the RMSD reported in the Reply Log.

The final RMSD (Root-Mean-Square deviation) over all residue pairs is a measurement of the average distance between two sets of atoms. In the macro-complex context, are the atoms of superimposed structures. This means that one of the two structures has been rotated and translated to minimize the distance between the two of them. It may be used as a measure of similarity between the two structures, thus the lower is the RMSD, better is the model.

Analysis of the proposed macro-complexes examples

4G83. Crystal Structure of p73 DNA-Binding Domain Tetramer bound to a Full Response-Element

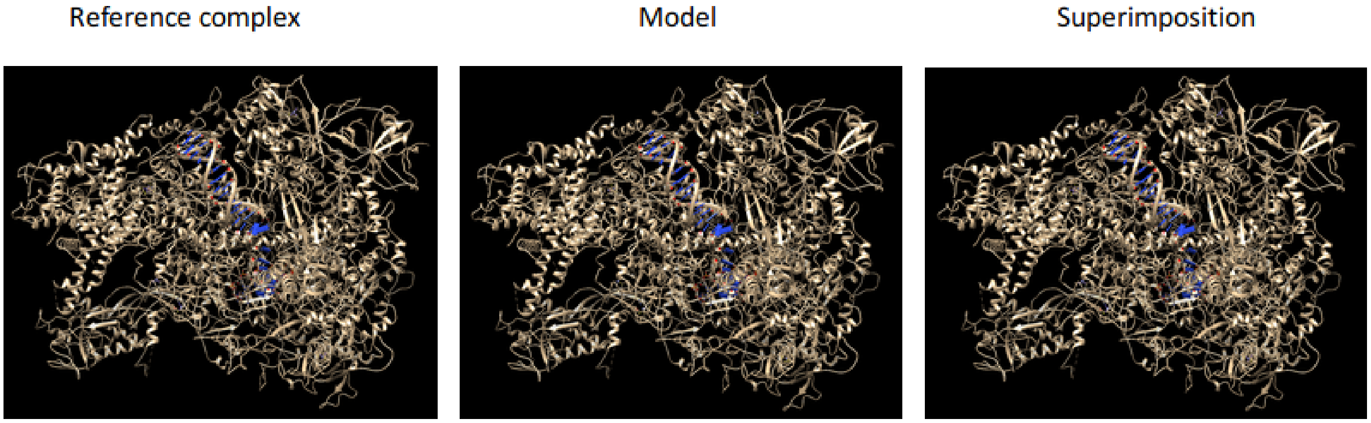
p73, P53-Like Transcription Factor or P53-Related Protein, has been considered a tumor suppressor because of its structural resemblance to p53. It is a DNA binding transcription factor involved in cell cycle arrest, regulation of transcription DNA templated and induction of apoptosis. This complex has four chains, with protein and DNA structure. Chains A and B correspond to the tumor protein p73; and chains E and F to DNA.


Reference complex
Model
Superimposition

Observations: The obtained complex was an homodimer + 2 nucleic acids (final stoichiometry: A2 + 2 nucleic acids), which is in accordance with the real stoichiometry. RMSD between 198 pruned atom pairs is 0.000 angstroms; (across all 198 pairs: 0.000).

5FJ8. Cryo-EM structure of yeast RNA polymerase III elongation complex at 3. 9 A

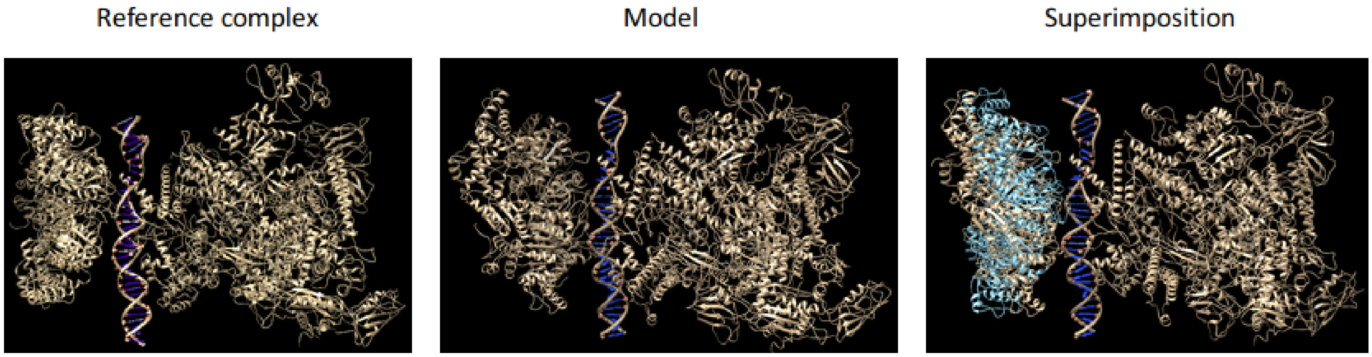
RNA polymerase III transcribes DNA to synthesize small RNAs as ribosomal RNA and transfer RNA. This macro-complex presents the cryo-electron microscopy structures of the *Saccharomyces cerevisiae* Pol III. Chain R corresponds to RNA; chains S and T to DNA (template and non-template); and chains A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P and Q to RNA polymerase III subunits.


Reference complex
Model
Superimposition

Observations: The obtained complex was an heterodimer + 3 nucleic acids (final stoichiometry: A1E1G1H1I1P1F1O1K1B1D1C1M1J1L1N1Q1 + 3 nucleic acids), which is in accordance with the real stoichiometry. RMSD between 1422 pruned atom pairs is 0.000 angstroms; (across all 1422 pairs: 0.000).

5NSS. Cryo-EM structure of RNA polymerase-sigma54 holoenzyme with promoter DNA and transcription activator PspF intermediate complex

RNA polymerase it's a DNA binding protein which transcribes DNA templated to synthesize RNA. This macro-complex includes sigma54 related to the RNA polymerase which is a protein needed for initiation of transcription in bacteria. It also includes Psp operon transcriptional activator. Chains H and I correspond to DNA; chains F, G, J, K, L and N correspond to Psp operon transcriptional activator; chains A, B, C, D and E to RNA polymerase subunits; and chain M corresponds to RNA polymerase sigma-54 factor.


Reference complex

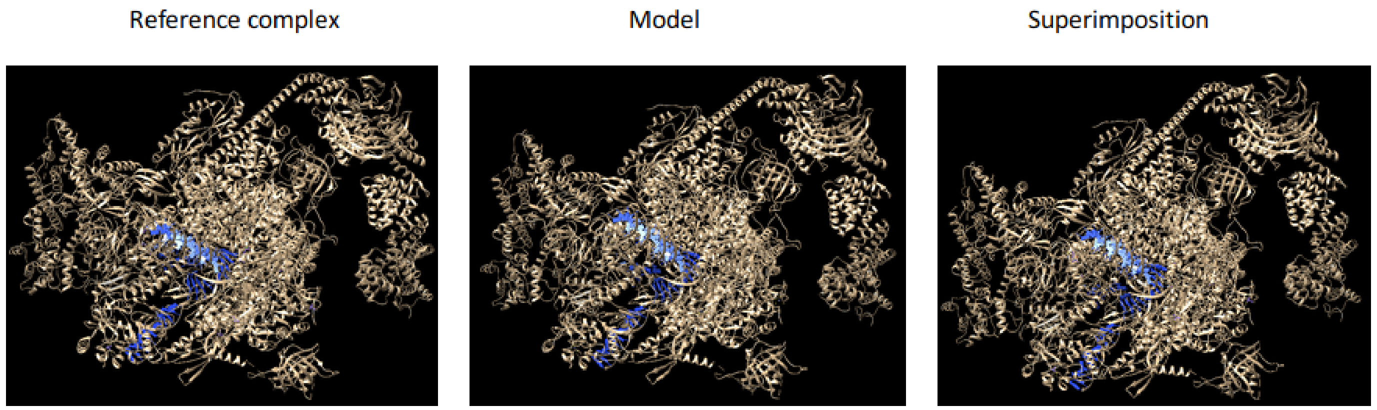
Model

Superimposition

Observations: The obtained complex was an heterodimer + 2 nucleic acids (final stoichiometry: A2C1D1M1E1F6 + 2 nucleic acids), which is in accordance with the real stoichiometry. RMSD between 1340 pruned atom pairs is 0.000 angstroms; (across all 1340 pairs: 0.000).

6GMH. Structure of activated transcription complex Pol II-DSIF-PAF-SPT6

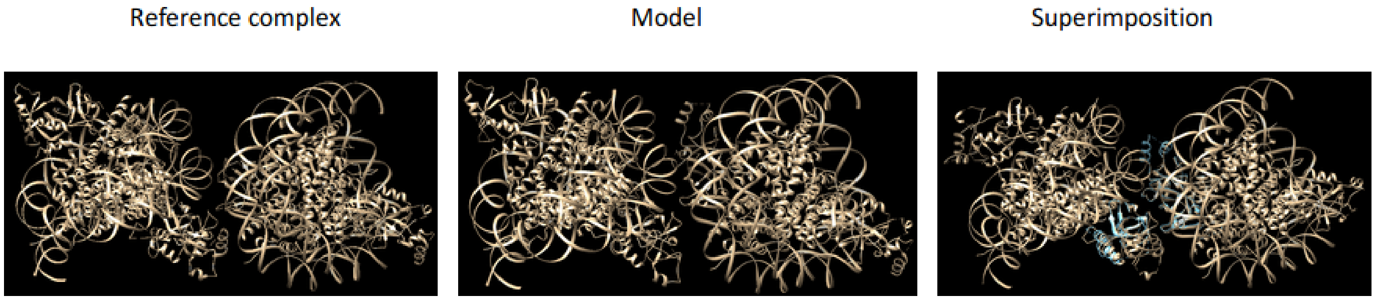
RNA polymerase II calatyzes the transcription of DNA to synthesize mRNA and most snRNA and microRNA. This macro-complex also includes the transcription elongation factors SPT4, SPT5 and SPT6. The SPT4-SPT5 complex mediates both activation and inhibition of transcription elongation, and plays a role in pre-mRNA processing, and SPT6 plays a role in maintenance of chromatin structure during RNA polymerase II transcription elongation. It also includes: PAF1 complex, which is a RNA polymerase II associated factor and includes some components as protein CTR9 homolog, the protein LEO1, the cell division cycle 73 protein (CDC73) and the WD repeat-containing protein 61. Chain P corresponds to RNA; chains N and T to DNA; chains A, B, C, D, E, F, G, H, I, J, K and L to RNA polymerase II subunits; chain Q to protein CTR9 homolog; chain U to protein LEO1; chain V to PAF1; chain W to WDR61; chain X to CDC73; and chains Y, Z and M to transcription elongation factors SPT4, SPT5 and SPT6 respectively.



Observations: The obtained complex was an heterodimer + 3 nucleic acids (final stoichiometry: A1Z1E1I1G1M1B1F1C1K1H1U1V1J1L1D1Q1W1X1Y1 + 3 nucleic acids), which is in accordance with the real stoichiometry. RMSD between 1441 pruned atom pairs is 0.000 angstroms; (across all 1441 pairs: 0.000).

6OM3. Crystal structure of the Orc1 BAH domain in complex with a nucleosome core particle

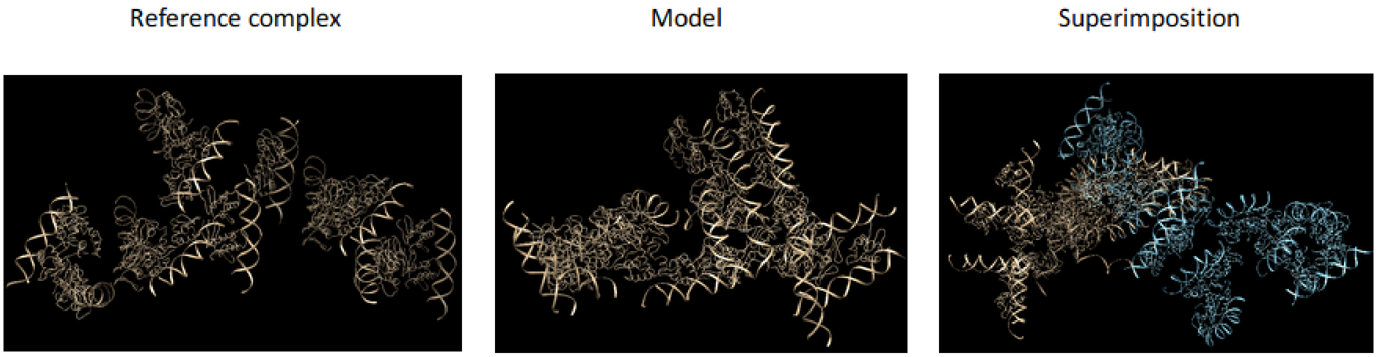
Origin recognition complex subunit 1 (ORC1) is a component of the origin recognition complex (ORC) that binds origins of replication. ORC is required to assemble the pre-replication complex necessary to initiate DNA replication. This macro-complex presents the structure of the yeast ORC1BAH domain bound to the nucleosome core particle. The nucleosome is the basic structural unit of DNA packaging in eukaryotes and consists of a segment of DNA wound around 8 histones. Chains A, E, M, Q, B, F, N, R, C, G, O, S, D, H, P, T correspond to different histones; chains K, L, W and X correspond to ORC1; and chains I, U, J and V correspond to DNA.



Observations: The obtained complex was an heterodimer + 4 nucleic acids (final stoichiometry: A4B4K4C4D4 + 4 nucleic acids), which is in accordance with the real stoichiometry. RMSD between 195 pruned atom pairs is 0.000 angstroms; (across all 195 pairs: 0.000).

3T72. PhoB(E)-Sigma70(4)-(RNAP-Betha-flap-tip-helix)-DNA Transcription Activation Sub-Complex

PhoB is a two-component response regulator that activates transcription by interacting with the rpoD, also called sigma70, subunit of the E. coli RNA polymerase. Chains C, G, K, O, T ,X, 2, 6, a, e, i, m, D, H, L, P, U, Y, 3, 7, b, f, j and n correspond to DNA; chains A, B, E, F, I, J, M, N, R, S, V, W, Z, 1, 4, 5, 8, 9, c, d, g, h, k and l correspond to protein PhoB; and chains o and q to RNA polymerase sigma factor rpoD.

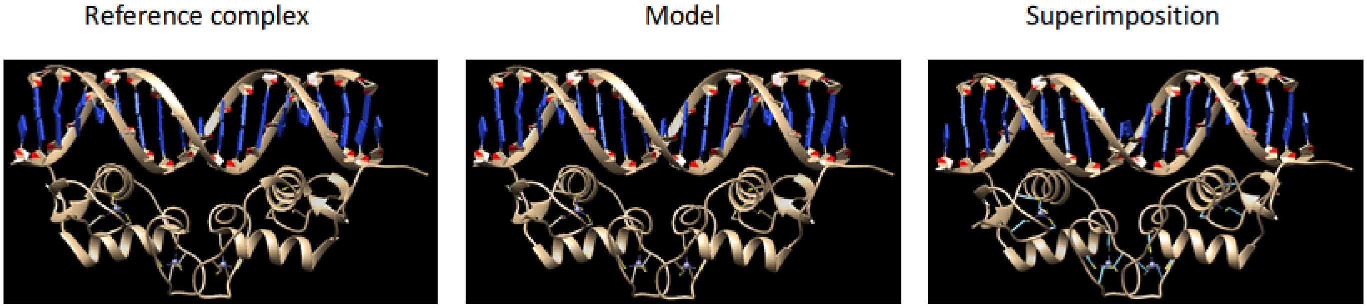


Observations: The obtained complex was an heterodimer + 24 nucleic acids (final stoichiometry: 1(24)o(2) + 24 nucleic acids). RMSD between 98 pruned atom pairs is 0.354 angstroms; (across all 102 pairs: 0.876).

Human Progesterone Receptor:

2C7A. Structure of the progesterone receptor-DNA complex

Progesterone receptor is a DNA and Zinc Ion binding transcription factor, that regulates the transcription of DNA templated in the cell nucleus. Chains A and B correspond to progesterone receptor; and chains C and D to DNA.

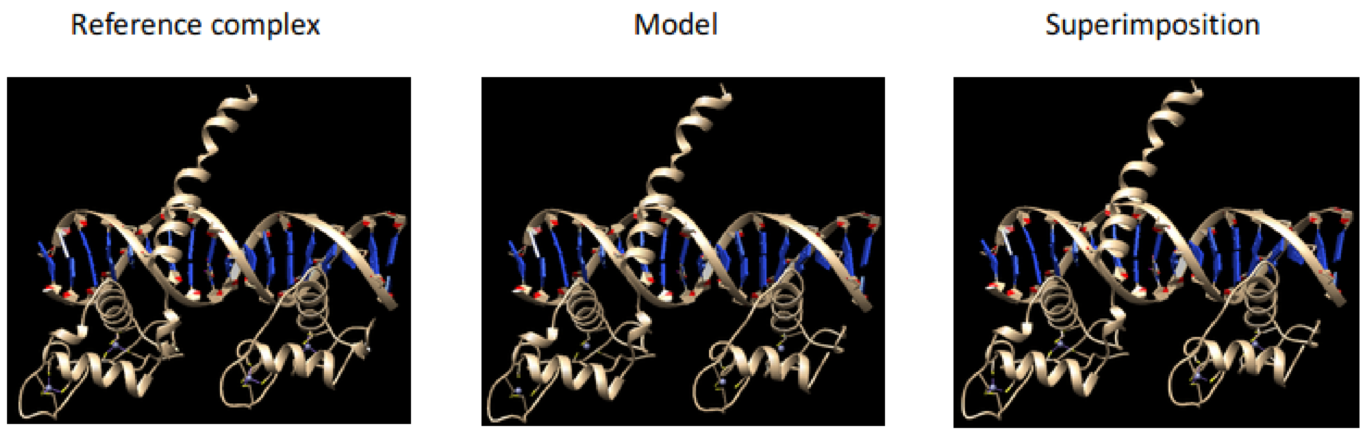


Observations: The obtained complex was an homodimer + 2 nucleic acids (final stoichiometry: A2 + 2 nucleic acids), which is in accordance with the real stoichiometry. RMSD between 78 pruned atom pairs is 0.000 angstroms; (across all 78 pairs: 0.000).

Human Thyroid Receptor:

2NLL. Retinoid X receptor-thyroid hormone receptor DNA-binding domain heterodimer bound to thyroid response element DNA

Retinoic acid receptor and thyroid hormone receptor are DNA and Zinc Ion binding transcription factors, that regulate the transcription of DNA templated, and their function takes place in the cell nucleus. Chain A corresponds to retinoic acid receptor; chain B to thyroid hormone receptor; and chains C and D to DNA.



Observations: The obtained complex was an heterodimer + 2 nucleic acids (final stoichiometry: A1B1 + 2 nucleic acids), which is in accordance with the real stoichiometry. RMSD between 103 pruned atom pairs is 0.000 angstroms; (across all 103 pairs: 0.000).

Limitations

The limitations of the program are the following:

- The algorithm is based on a recursive function, which is highly time consuming. In order to increase efficiency, when trying to add new chains to the complex, the program select the first possible option, instead of creating a different model for each option and then rejecting some of them or just showing the best ones. This limitation is evident in the complex 5nss, where a part of the structure is not builded as in the real complex.
- When the optional stoichiometry argument is passed to the script, the user is given a message and is asked to input information via terminal. This might cause issues if the user decides to redirect the standard output channel to a file as the message won't appear in the terminal and it would appear that the script is running, even though it is waiting for the user to enter chain information.
- No optimization of the model is performed via MODELLER
- Small compounds, such as ligands, hormones or metabolites, are not handled.
- The structure of 37t2 is not well build. This could be explained by the fact that some parts of this macrocomplex are separated by a distance bigger than 8 Å, which is the maximum distance allowed by our module interaction in order to consider that two chains are interacting. Therefore, in this case, the program does not properly work due to the lack of information of the interactions.
- Regarding the real incomplete examples, the program is able to rebuild properly the structure from the already elucidated part of the complex. However, the program is not able to use the information of different domains that are part of a single chain in order to join them and build a new complex.

Requirements

Biopython is needed for superimpositions, sequences alignment and obtaining the input files.
If user installs via *pip* requirements are automatically checked and installed if necessary.

Bibliography

- Adam Hospital, Pau Andrio, Carles Fenollosa, Damjan Cicin-Sain, Modesto Orozco, Josep Lluís Gelpí. MDWeb and MDMoby: an integrated web-based platform for molecular dynamics simulations. *Bioinformatics,* *2012, 28(9):1278-1279.*

- UCSF Chimera--a visualization system for exploratory research and analysis. Pettersen EF, Goddard TD, Huang CC, Couch GS, Greenblatt DM, Meng EC, Ferrin TE. *J Comput Chem. 2004 Oct;25(13):1605-12.*