ProtNAff: Protein-bound Nucleic Acid filters and fragment libraries

Antoine Moniot, Yann Guermeur, Sjoerd Jacob de Vries, Isaure Chauvot de Beauchêne

Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

OXFORD

Subject Section

ProtNAff: Protein-bound Nucleic Acid filters and fragment libraries

Antoine Moniot [1], Yann Guermeur [1], Sjoerd Jacob de Vries [2,3,*] and Isaure Chauvot de Beauchene [1,*]

[1] LORIA (CNRS - INRIA - UniversitÃ© de Lorraine), Nancy, 54000, France
[2] Ressource Parisienne en Bioinformatique Structurale (RPBS), Paris, France
[3] BFA, CNRS UMR 8251, INSERM ERL U1133, Paris, France.

* To whom correspondence should be addressed: sjoerd.de-vries@inserm.fr, isaure.chauvot-de-beauchene@loria.fr

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Atomistic models of Nucleic Acids (NA) fragments can be used to model the 3D structures of specific protein-NA interactions and address the problem of great NA flexibility, especially in their single-stranded regions. One way to obtain relevant NA fragments is to extract them from existing 3D structures corresponding to the targeted context (e.g. specific 2D structures, protein families, sequences) and to learn from them. Several databases exist for specific NA 3D motifs, especially in RNA, but none can handle the variety of possible contexts.
Results: This paper presents protNAff, a new pipeline for the conception of searchable databases on the 2D and 3D structures of protein-bound NA, the selection of context-specific (regions of) NA structures by combinations of filters, and the creation of context-specific NA fragment libraries. The strength of this pipeline is its modularity, allowing users to adapt it to many specific modeling problems. As examples, the pipeline is applied to the quantitative analysis of (i) the sequence-specificity of trinucleotide conformations, (ii) the conformational diversity of RNA at several levels of resolution, (iii) the effect of protein binding on RNA local conformations, and (iv) the protein-binding propensity of RNA hairpin loops of various lengths.
Availability: The source code is freely available for download at URL https://github.com/isaureCdB/protNAff. The database and the trinucleotide fragment library are downloadable at URL https://zenodo.org/record/6483823#.YmbVhFxByV4.
Contact: isaure.chauvot-de-beauchene@loria.fr sjoerd.de-vries@inserm.fr

1 Introduction

The uses of atomistic models of protein-nucleic acids (NA) assemblies are numerous. For instance, they are the keystone of rational drug design targeting those complexes. The knowledge of the 3D structure of the interaction between a NA and a protein makes it possible to design a therapeutic NA with a better binding affinity to the protein. 3D structural modeling can also play a key role in understanding the effect of a mutation on the protein, for example when it changes the recognized NA sequence (Masaki et al. (2019)). The Protein Data Bank (PDB) currently contains 14,188 assemblies of DNA/RNA with proteins, including a wide variety of biological systems. Ribosomes account for about 10 percent of all those assemblies, leaving about 12,780 non-ribosome nucleotide chains in the PDB. This is much smaller than the total number of protein chains (about 179,885) in the PDB. Still, about 16 percent of all proteins in the human genome (from UniProt) contain at least one RNA-binding domain; the most common one, the RRM domain, is present in 2% of all proteins in the human genome (Maris and C. Dominguez (2005)). This means that DNA/RNA are under-represented in the PDB compared to proteins.
Undoubtedly, the cause of this under-representation is the great flexibility of nucleic acids (Fulle and Gohlke (2008)), which also poses multiple challenges for 3D structural modeling. First, as argued above, there are fewer experimental structures to work with, because flexibility causes experimental difficulties. In addition, the great flexibility of NA must be taken into account for their modeling and can make it difficult (Jones (2016)).

In particular, it enlarges the conformational space to model: whereas for proteins, general fragment libraries exist that describe essentially the entire possible conformational space (Bhattacharya et al. (2016)), such general libraries are absent for nucleic acids, for good reasons. Nucleic acids in general adopt much more context-specific conformations. For DNA, there is often considerable induced fit upon binding to a protein (Mias-Lucquin and de Beauchene (2021)). For purely single-stranded RNA, the unbound conformation is either disordered or in another secondary structure, and becomes defined only upon binding. This necessitates the creation of NA assembly databases and fragment libraries that are specific to a given context (e.g. single-stranded NA, contact with a protein, experimental method of resolution, etc).

An example of a very specific context is the subset of RNA structures from crystallography only, with at least 2Å of resolution, with an RNA hairpin loop of at least 5 nucleotides, at least 70% of which are in contact with a protein. This context was used to create a benchmark for a specific hairpin docking method in (Moniot et al. (2019)).

There exist already many databases and libraries dedicated to DNA structures (Zheng et al. (2010); Sagendorf et al. (2020)). Those databases are made to visualize and search conformational structures on the DNA. For RNA structures, there is a variety of context-specific tools. **RNANet** (Becquey et al. (2021)) is a database that provides a large number of RNA geometric and sequence descriptors. The focus is on individual RNA chains extracted from crystal structures. Another database is **RNA CoSSMos 2.0** (Richardson et al. (2020)) that supports a large variety of loop motifs to be queried. Neither database accounts for interaction with a protein. Each of these tools does some specific task on RNA structures, but none of them can handle both DNA and RNA, nor create fragment libraries.

In principle, each new context and new usage requires a new database. Furthermore, new data are being made available all the time. Therefore, the creation of new and updated NA databases will continue to be necessary to advance the field. Yet building such databases represents a great amount of work, much of it being often manual as there is a lack of a generic tool to build NA databases in an automated manner.

The **protNAff** (protein-bound Nucleic Acids filters and fragments) pipeline presented in this article proposes a solution to meet the shortcomings. It consists of correcting and parsing protein-NA structures from the PDB (partially using tools from the ATTRACT suite (de Vries et al. (2015))), extracting a large set of their structural features (partially using the DSSR tools of the 3DNA suite (Lu and Olson (2003))), and organising these data into a JSON file. The output can be processed by any programming language, to apply customised filters. The filters make it possible to identify structures based on a complex set of criteria, extract their regions of interest, and make context-specific libraries of fragments with different levels of redundancy. It also allows performing automated analysis of the effect of a context on NA local features, such as the diversity of fragment conformations or their type of interactions with proteins. In this paper, as a proof of concept, a set of filters written in Python is proposed, corresponding to different contexts, that can be easily adapted to the user's needs.

ProtNAff currently specializes in the analysis at the level of trinucleotide fragments. One particular strong point is the ability to modulate the resolution of the conformational space. For the statistical analysis of local conformation features in different contexts and their biological interpretation, the proper threshold for clustering (it means the radius of the clusters) the NA fragments is essential to detect context-specific features. As an example, to detect if some local NA conformations are specifically induced by binding to a given protein family, the chosen clustering threshold must be high enough to create family-specific clusters, but low enough for those clusters to contain fragments from a statically sufficient number of structures bound to different members of the family. In addition, the granularity of conformational space has implications for the use of NA fragment libraries in 3D modeling. Here, the fragment clustering threshold must balance the desired level of accuracy and the maximal size of the library (i.e., the number of conformations), which depends on the modeling process and its application. As an example, for the sampling step in fragment-based docking with trinucleotides, several options are possible. The first one is to use a small number of very different fragments, corresponding to a high clustering threshold, and add explicit flexibility (e.g. molecular dynamics) to cover the conformational landscape around each fragment. The second one is to do rigid sampling with conformers much closer from each other (thus a larger library), corresponding to a low clustering threshold, in order to obtain an exhaustive enough sampling of the conformational landscapes.

This versatility allows protNAff to cover a very large part of the multiple possible contexts mentioned above. Moreover, the versatility of the tool, by avoiding predefined strict criteria, places it in a better position to leverage the new types of structural data or the new features of interest that could become available in the near future. One strength of the tool is that the resulting database is stored locally, so that the user has control on the database and can use it as he sees fit. Moreover, the database can be easily updated to incorporate new structures deposited in the PDB. The different use cases can be implemented collaboratively over time, with the code accessible on GitHub.

In this paper, the creation of a database, the selection of a subset of structures, and the building of fragment libraries are explained, and the different steps of each process are detailed in Fig. 1.

The results of four quantitative analyses performed with protNAff are presented here as examples. Each example uses a complex combination of filters. We focus on RNA because RNA structures are more diverse in the PDB than DNA structures, but the same analyses have been applied to DNA as well (results are presented in Supplementary). The four examples are as follows. First, the sequence-specificity of trinucleotide conformations at 1Å resolution is measured for each purine/pyrimidine motif. Second, the conformational diversity of trinucleotides is measured at different levels of resolution. Third, the fraction of existing RNA local conformations that are specifically induced by protein binding and do not exist in unbound RNA regions is measured for different RNA sequences and 2D structural states. Fourth, the protein-binding propensity of RNA hairpin loops is compared for different loop lengths.

The database and library (clustered at 1Å) created by protNAff for those analyses are made available for download at URL https://zenodo.org/record/6483823#.YmbVhFxByV4, allowing to use and analyse them without running the full pipeline.

2 Methods

2.1 Creation of the database

The creation of a database is based on several main steps. To initiate the process, the user gives a list of pdb identifiers. This list can be obtained as a result of a query on the PDB site, which allows a first filtering by the user. The `.pdb` files are downloaded and then edited.
The experimental method to obtain the structure, and the resolution if available, are extracted (by parsing the description in the pdb file). The water, ions, and other small ligands are removed. Missing atoms in nucleotides or amino acids are detected using ATTRACT `aareduce.py`

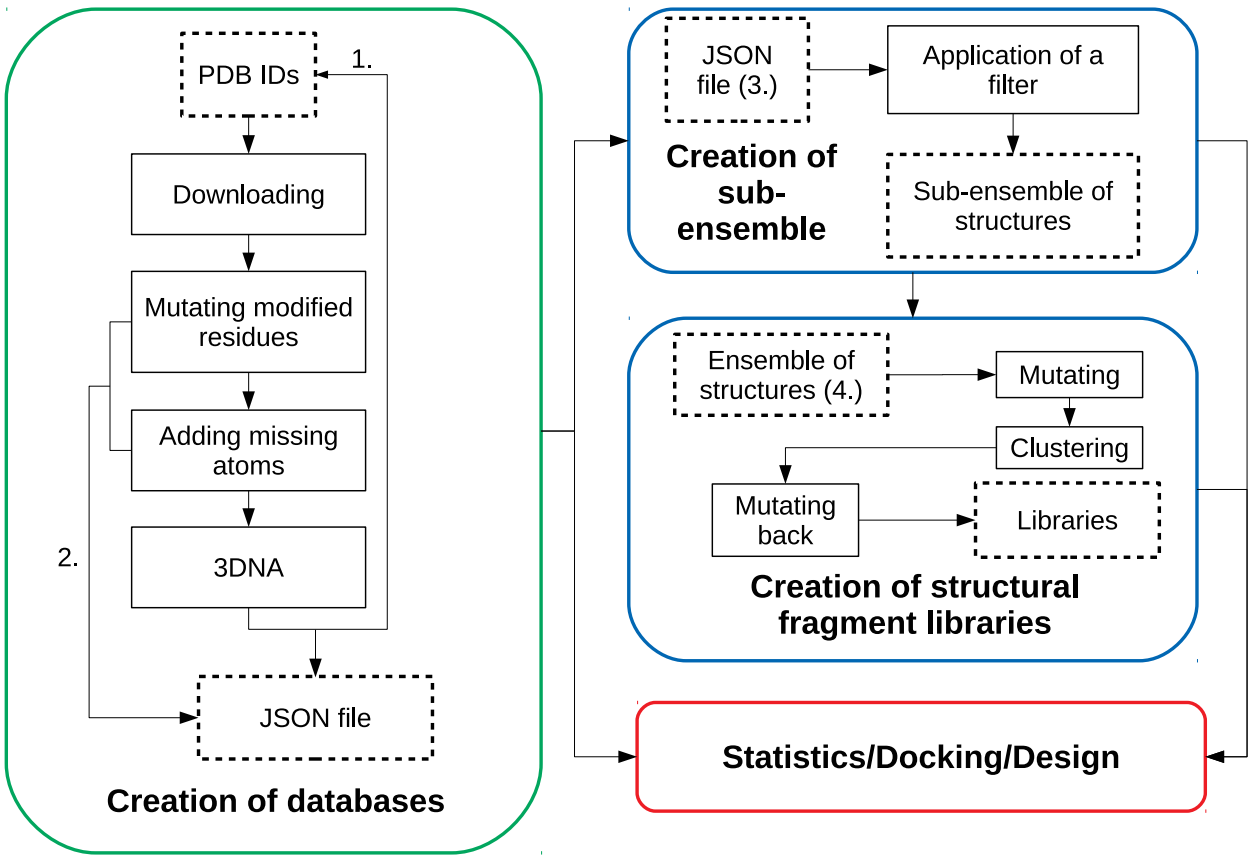

Fig. 1. This global diagram presents the pipeline and its most important steps. Each output - database, subsets, fragments - can be useful independently: search for particular examples of complexes, statistical analyses on patterns, fragment-based modeling, etc. The doted boxes are the input and output files. 1. Every steps are done for each pdb ids. 2. The data about modifications made on the pdb files are stored in the database. 3. The JSON file is the one obtained as a database. 4. The user will create his ensemble of structures to make structural fragment libraries.

tool. Nucleotides that contain less than 4 atoms are discarded, others are kept to be fixed by adding the missing atoms. For an amino acid, the pdb2pqr (Dolinsky et al. (2007)) library is used. For a nucleotide, a mononucleotide library is aligned on the existing atoms using the `fit.py` tool of ATTRACT, and the atoms of the nucleotide that produces the best alignment (lowest residual RMSD) are added. One structural library of mononucleotides per RNA/DNA base is created, by extracting all nucleotides from the pdb, removing those non-canonical or with missing atoms, clustering them at 0.3Å RMSD, eliminating clusters with less than 10 members, and keeping a prototype in each cluster. Information on which nucleotides of a structure had missing atoms and in which part (phosphate, sugar, base) is stored in the database. This allows a user to choose how to treat those nucleotides, for instance, to discard or keep them in a fragment library depending on constraints of exhaustiveness or correctness. Modified nucleotides with additional atoms are also detected using the ATTRACT `aareduce.py` tool. If possible, they are transformed into the closest (highest number of common atoms) canonical nucleotide by removing a few atoms, according to a manually created mapping (`protNAff/data/[r/d]nalib/mutate.list`). They are discarded if they are too far from any canonical nucleotide. The final database stores a list of which nucleotide had been canonized and from which original modified nucleotide type. This allows the user to keep either all fragments (to get closer to exhaustiveness) or to keep only genuine fragments (to get closer to correctness), depending on the applications of the library. The protNAff script `filter_no_modified.py` is provided to discard the modified fragments.

A re-numbering of the nucleotides starting from 1 is applied. If there are alternative positions for some residues/atoms, several `.pdb` files are created, one for each set of alternate positions. For each part (phosphate, sugar, base) of each nucleotide, the minimal distance of any heavy atom to a protein or cofactor heavy atom is computed, and saved if below a chosen threshold (default 5Å). The DSSR tool of the 3DNA package is applied on the modified `.pdb` files. This provides exhaustive information on the nucleic acid 2D and 3D structure, much of which is integrated into the database. This piece of information is then parsed and converted from structure level to nucleotide level. For instance, from a "nucleotides 5 to 15 make a stem-loop" statement from DSSR output is extracted a "nucleotide 5 is at position 1 in an 11-nucleotides stem-loop" statement to be stored in our database. The database is placed in a JSON file, for easy parsing with the preferred programming language.

2.2 Content of the database

The first output of the protNAff pipeline is a JSON file containing information per structure (for instance crystallographic resolution), per nucleic acid chain (for instance positions of breaks in the backbone) and per nucleotide (for instance: H-bonds with the protein). The second output of protNAff is a library of fragments (consecutive trinucleotides in the current implementation), consisting of fragments coordinates (in

npy-arrays (Harris et al. (2020)) or pdb format) and metadata/structural information per fragment, stored in another JSON file (pdb code of the original structure, cluster index, missing atoms in initial structure, etc). A complete list of the per-structure and per-fragment data provided by protNAff is given in Supplementary. The combination of the two JSON files containing information one per structure and the other one per fragment allows to compute all kinds of statistics on the library. Some examples are presented in the Results section.

2.3 Customized filters

The user can apply filters on the database for the creation of ensembles of structures corresponding to a given context. The uses of these ensembles can be multiple and therefore depend on the user. For instance, they can be used to create reference sets of complexes to test a docking method. This step is the least automated one, so as to let as much freedom as possible to the user. It consists of writing a script that browses the JSON file and collects data of interest.
An explanation notebook is provided (`filters/explanation_filters.ipynb`) on how to use the JSON files to create filters. The example given at the end of the notebook select the base-paired nucleotides (WC pairing) that are in contact with a protein.

2.4 Fragment libraries

ProtNAff allows the creation of structural fragment libraries. These libraries can be used as building blocks for structure prediction (RNA modeling, docking, design) or to compute statistics on particular assemblies. To create these libraries, the structures of interest are selected and their fragments are identified via the JSON file. These fragments are extracted from the cleaned pdb structures, pooled by sequence, and clustered to obtain a set of prototypes.
In the case of our fragment-based docking method (de Beauchene et al. (2016)), the fragments of interest are the consecutive trinucleotides. It is however possible with some modifications to create libraries of other structures, for instance, dinucleotides, or double-stranded helices.

2.4.1 Systematic mutations

To artificially increase the number of conformations for each sequence, all combinations of mutations are computed to transform purines (R), i.e., the Guanines into Adenines and vice-versa, and pyrimidines (Y), i.e., Uracils into Cytosines and vice-versa. This rests on the assumption that such mutations, which add or remove one heavy atom, have a negligible impact on the overall structure of the trinucleotide. We verified this assumption by the fourth analyse given as example in this study. To optimise the computation, all bases are first mutated in A or C, the clustering is applied, then the clustered fragments are mutated back by all combinations of U/T to C and A to G mutations. For trinucleotide fragments, each trinucleotide is mutated into $2^3 = 8$ new sequences, so the size of sets is multiplied by 8 approximately (because of redundant conformations).

2.4.2 Clustering

Once all the fragments for the chosen structures have been retrieved and extracted in pdb format, clustering is performed. The purpose of this clustering is to obtain representatives of all the existing fragments (observed or not).

The user has the possibility to choose the radius of the clusters, which allows more flexibility. For instance, for docking a trinucleotide using flexibility modeling, one may want a large radius, say 3Å, since the docking will explore new conformations around the initial fragments. However, for rigid docking of trinucleotides, one may want to have a smaller radius, say 1Å, to cover the conformational landscape with high precision (see Section 3.2).

Clustering is carried out several times with different radii: a first clustering with a 0.2Å RMSD radius is applied to remove redundancies (in particular fragments from two biological assemblies from the same crystal structure), then the non-redundant fragments are clustered first with a 1Å then 3Å RMSD radius.

The clustering algorithm is as follows. The initialization is performed by randomly choosing a fragment as the 1st cluster prototype. Then, for each fragment, the distance (RMSD) to each of the prototypes of the current set is measured. If any of these distances is less than the chosen radius, then the fragment is assigned to the nearest cluster. Otherwise, it is added to the set of prototypes. Once this first pass on the whole pool of fragments is completed (i.e., once all the prototypes have been obtained), a second pass is performed: fragments that are not prototypes are reassigned to the nearest cluster.

A second clustering method (Moniot et al. (2022)) is implemented in protNAff. It is a hierarchical agglomerative clustering characterized by its linkage function that takes in input two clusters and returns the radius of the minimum enclosing ball. The resulting prototypes are not input fragments but the centers of the balls. If the user needs a library made up of a subset of the initial fragments, the first clustering method is the one to favor, otherwise the second one is better as it produces fewer clusters. A comparison between those two clustering methods applied to trinucleotide fragments can be found in the `clustering_comparison.ipynb` notebook.

3 Results/Instances

The list of pdb IDs given as input to obtain the results exposed in this Section is all pdb structures that contain protein chains and RNA chains but no DNA, where the resolution is less than 3Å or where the method is NMR. The ribosomes are removed from this database due to their size.

All the statistics and figures can be found in the notebook `figures_protnaff.ipynb` on the GitHub webpage of protNAff https://github.com/isaureCdB/protNAff. The fragment library can be downloaded at .

3.1 Sequence-specificity of RNA conformations

We explored to which extend trinucleotide conformations are sequence-specific for a given purine/pyrimidine motif. ProtNAff applies all possible purine to purine and pyrimidine to pyrimidine mutations, and clusters the fragments separately for each motif (see the Methods section). To answer our question, we analyzed the population of the 1Å clusters in terms of the original sequence of the fragments. The purpose was to assess if some conformations are specific to a given original sequence, meaning that the corresponding cluster contains only fragments with that original sequence. The details of this analysis can be found in the jupyter notebook called `sequence-specific_conformations.ipynb`, in the github repository.

For a cluster of n fragments, the probability to have a pure cluster (containing only one sequence) decreases with n. In our clusters at 1Å RMSD, for all R/Y-R/Y-R/Y motifs together, we found less than 5% pure clusters for n>=3 (32/715), less than 3% (11/461) for n>=4 and less than 2% (6/346) for n>=5. Thus, the proportion of single-sequence clusters remains very low, meaning that the large majority of conformations (defined at 1Å) are not specific for one sequence for a given

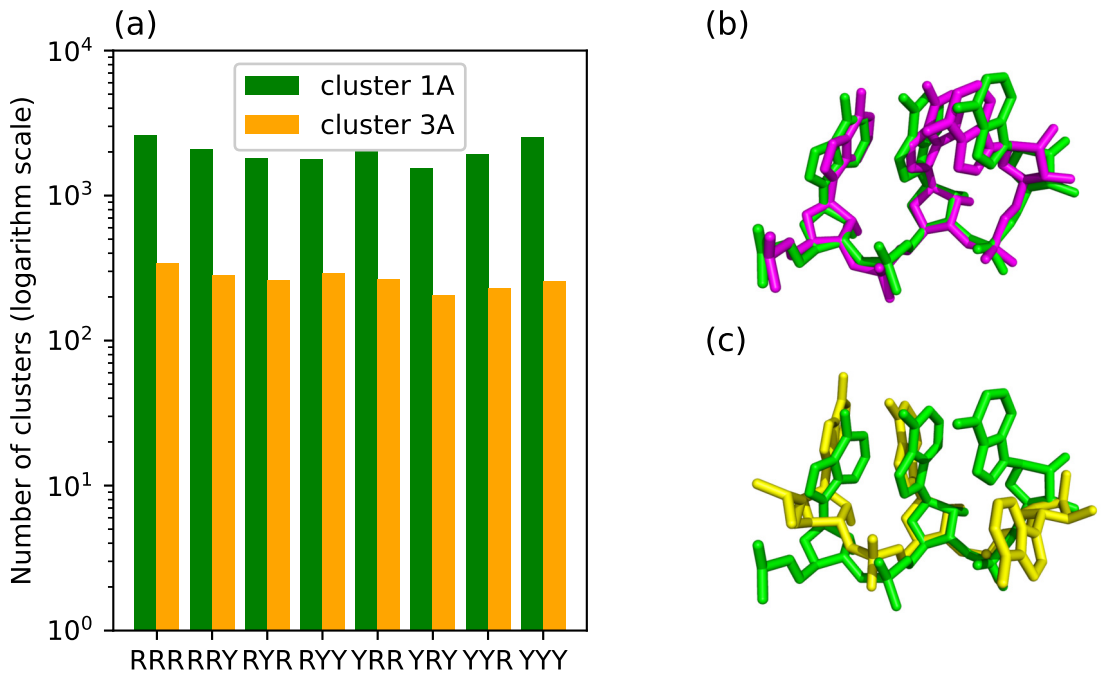

Fig. 2. Clustering on RMSD with different cutoffs: (a) Number of clusters depending of the width of the clusters (1Å or 3Å), in logarithmic scale; Random examples of alignment of two different trinucleotides at 0.9Å RMSD (b) or 2.5Å RMSD (c), to illustrate the conformational diversity inside clusters as a function of their width.

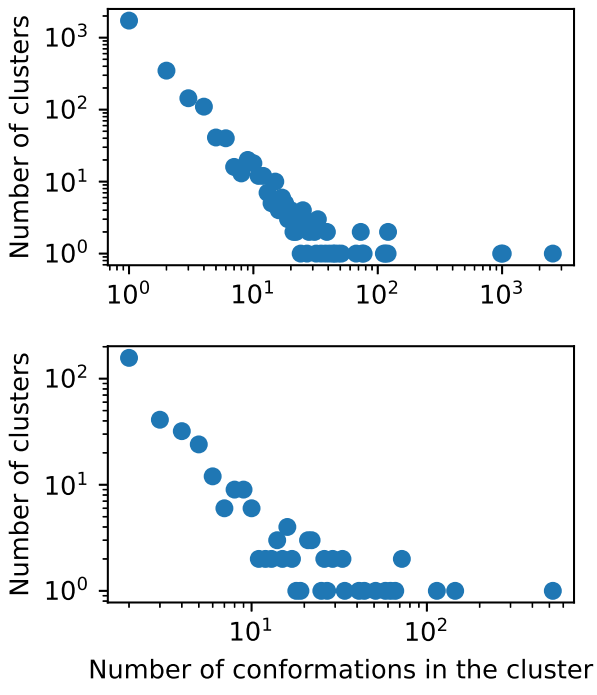

Fig. 3. The number of clusters containing a certain number of conformations after clustering at 1Å (top) and 3Å (bottom) for RNA. Both axes are in logarithm scale.

purine/pyrimidine motif.

Those results underline the relevance of performing systematic mutations when creating libraries of trinucleotides with maximal conformational exhaustivity, to the expense of allowing few unrealistic conformations (corresponding to the mutated while sequence-specific conformations). For application where avoiding unrealistic conformations in more important than exhaustivity, the mutated fragments should be filtered out. Note that such results do not apply to other types of fragments where base-pairing can occur, and for which the mutation pattern should be adapted. Such an example is available in the Github repository of protNAff (see create_helices_library.sh).

3.2 Conformational diversity of RNA fragments at different scales

Most usages of the database and libraries necessitate proper and context-specific handling of redundancies. At the level of the fragments, different clustering thresholds can modulate the granularity of the conformational space representation. Therefore, the implementation of protNAff allows the user to adjust the radius of the clusters depending on their purpose. This allows finding a balance between the maximum size of the library that can be handled by modeling tools and the desired precision of the discretization of the conformational landscape for that type of fragment. As an example, the number of clusters of trinucleotides obtained with a 1Å or 3Å RMSD threshold for the different pyrimidine/purine sequences is presented in Fig. 2.

The diversity of clusters at 1Å and 3Å is shown in Fig 3. Those figures indicate the number of clusters of a certain size. The distribution is similar for both clustering thresholds. The difference is that there are fewer but larger clusters at 3Å than at 1Å.

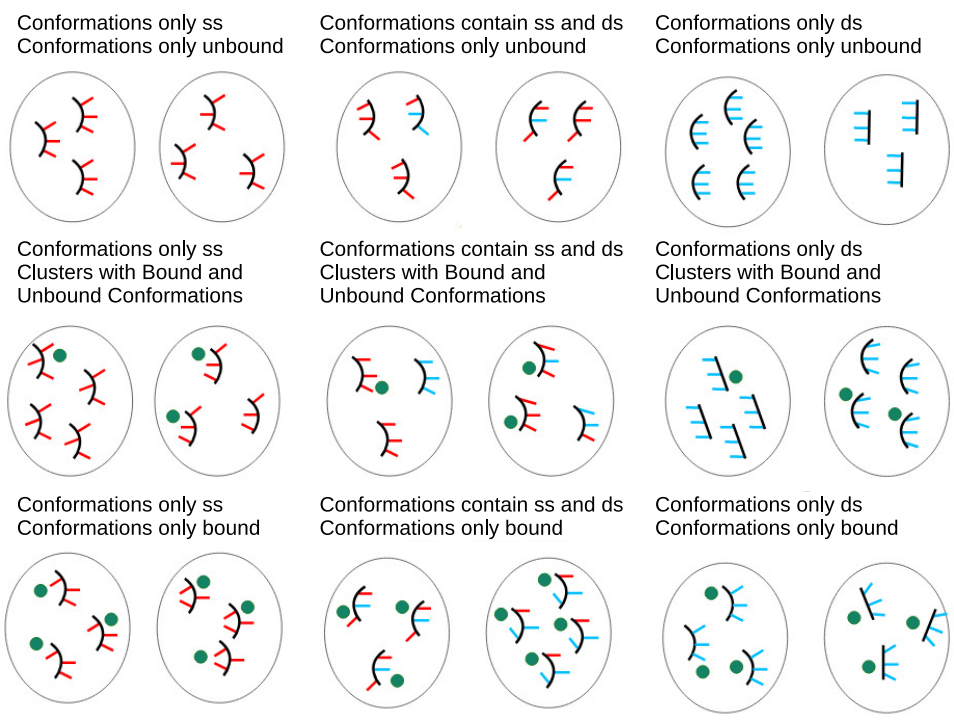

Fig. 4. Definitions for single-stranded/double-stranded/mixed and bound/unbound/mixed clusters. Single-stranded (ss) means that the 3 nucleotides are unpaired, double-stranded (ds) means that the 3 nucleotides are base-paired. Bound means that at least one nucleotide is in contact (distance inferior to 5Å) with the protein. Unbound means that there are no contact with the protein.

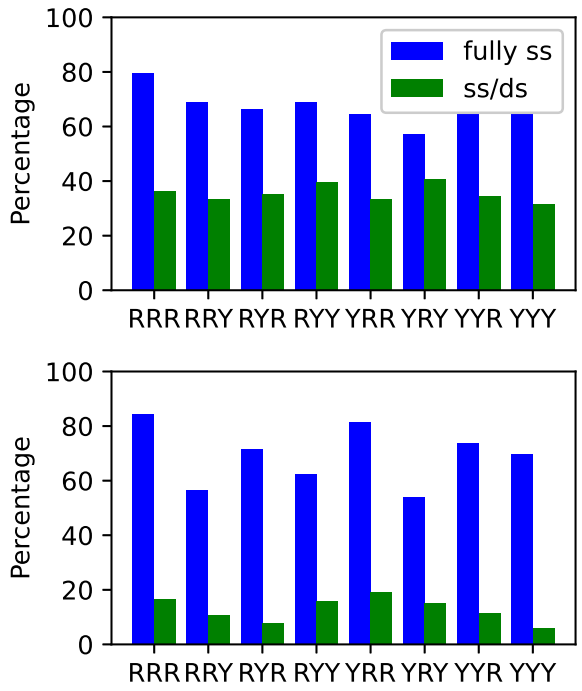

Fig. 5. Percentage of fully single-stranded conformations induced by the contact with the protein and the conformations that are not single-stranded. On the top, the selected conformations at 1Å, on the bottom, the selected conformations at 3Å.

3.3 RNA local conformations induced by protein binding

The hypothesis that RNA in contact with proteins can adopt specific local conformations compared to unbound RNA can be verified by clustering fragment conformations based on their pairwise structural difference (e.g. RMSD). If some clusters contain only fragments in contact with a protein, the corresponding conformation is specifically protein-induced, see Fig. 4 to understand the distinction between clusters. Moreover, one would expect that such protein-induced conformations are more numerous in ssRNA, which are less constrained by intra-RNA bonds. This can be assessed by distinguishing clusters made exclusively of ssRNA fragments. To verify both hypotheses, all overlapping RNA trinucleotides were extracted from the full set of protein-bound RNA in the PDB, a clustering with 1.0Å pairwise-RMSD threshold was made, and the number of clusters coming only from interface trinucleotides was counted, with a distinction between ssRNA and dsRNA, see Fig. 5. The clusters containing less than three conformations were considered as not representative enough and removed. Since 39% of all RNA fragments were not at the protein interface, we assumed that this set covers all possible unbound conformations of the RNA. The results are shown in Supplementary.

The low number of double-stranded clusters was expected, as the conformations encountered in helices can also be adopted by a fragment containing at least one single-stranded nucleotide (fragment considered as mixed here). The total number of fragments contained in the clusters is presented in Supplementary. The mixed/mixed clusters gather the majority of the fragments, which makes sense as they are the clusters containing the most frequent conformations. Around half of the clusters are made only of trinucleotides in contact with a protein, suggesting that these conformations are specifically (at 1.0Å resolution) induced by the contact with the protein. As expected, the percentage of conformations specifically induced by protein binding is higher for clusters made exclusively of single-stranded fragments, for all purine/pyrimidine motifs.

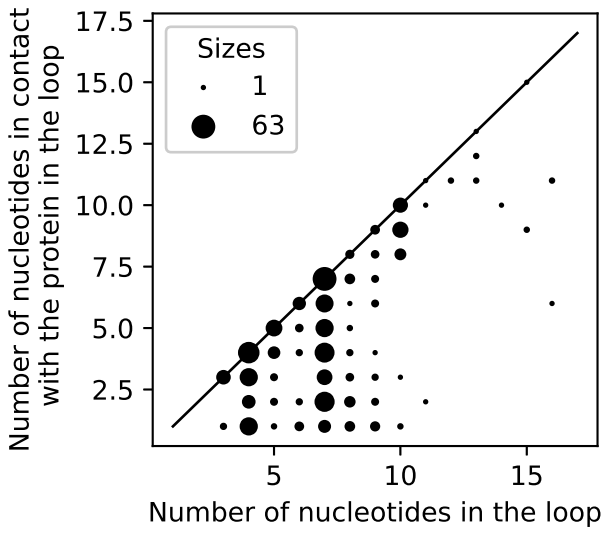

Fig. 6. Number of hairpin loop nucleotides in contact with the protein regarding the number of nucleotides in the loop. The size of the dots indicates the number of hairpins in this configuration, from 1 to 63.

A two-proportion Z-test was applied between the exclusive single-stranded population and the rest. For the eight possible motifs, the p-value was below 10^{-5} and the test was positive. Interestingly, the clusters fully single-stranded or fully double-stranded are either fully in contact or fully not in contact with a protein. That means that neither the fully single-stranded nor the fully double-stranded conformations exist in both bound and unbound states. In other words, the contact of such a fragment with a protein is always inducing a conformation different from those that can be observed in unbound RNA. Those results can be observed for clusters at 3Å (see Supplementary). One big difference is that there are almost no clusters fully double-stranded (only 2 clusters for the 8 sequences).

3.4 Size of interfaces in protein-bound RNA hairpin loops

We used a filter to understand if the size of the protein-RNA interface in hairpin loops correlates with the size of such loops.

The filter selects all the hairpin structures where the single-stranded loop is in contact with the protein (See filters/query_hairpin.py on GitHub). Here, the possibility is given to the user to choose the minimum/maximum size of the single-strand loop and of the double-strand helix. Using a minimum length of 3 nucleotides for the loop and 3 base-pairs for the helix, 1160 hairpins were obtained that belong to 437 pdb structures.

This filter allows us to investigate some protein-binding characteristics of RNA hairpins, such as the number of nucleotides in contact with the protein (single-stranded and double-stranded) for hairpins of different sizes. We investigated the outliers corresponding to a too long loop or a too small part in contact. For instance, the structure 6ZDQ has a 16-nt long loop with only 6 nt in contact. The structure 7KA0 has an 11-nt long loop with only 2 nt in contact, which can be explained by a strong interaction with another RNA chain. A plot comparing the size of the loops and the number of nucleotides in contact with the protein is presented in Fig. 6.

We found that most hairpins have a loop of 7 or fewer nucleotides and that a third of the hairpins has all nucleotides within 5Å from a protein. The longest loop has 16 nucleotides, with 11 of them bound. For loops with a length of up to 10 nucleotides, most of the possible numbers of protein-bound nucleotides are observed. Above 10 nucleotides, almost all loops have half or more of their nucleotides bound. At maximum, 9 nucleotides are not in contact with the protein, which means that for small loops, a high proportion can be away from the protein. In the case of long loops, the proportion of the loop not in contact becomes small. This may come from the fact that a longer single-stranded loop not in contact is too flexible to have a good resolution.

3.5 Analyses on DNA

We reproduced the same figures and statistics on DNA, all results are found in Supplementary. For DNA, the query on the PDB is: all complexes of protein/DNA (without RNA) with a resolution at most of 3Å. This query gives 1906 structures. Then the pipeline is applied to those structures and figures can be reproduced using the notebook `figures_dna_protnaff.ipynb`.

3.6 Memory and computational complexity

All times were obtained on 8 CPU Intel(R) Xeon(R) Silver 4114 CPU @ 2.20GHz, with 16Go of RAM.

The creation of the database is not a parallelized step. In our tests, downloading the RNA-protein complex files from the PDB for the creation of the database takes between 1 and 2 hours. The set of downloaded pdb files corresponds to a volume of 1.7 GB. Finally, updating the database, i.e. adding a new structure, takes a few minutes depending on the size of the structure.

Likewise, the creation of a library of trinucleotide fragments from the previous database corresponds to a 4h computation. On this step, the creation of clusters is parallelized : each sequence is associated with a thread. Finally, the computational time for the selection of a subset of structures will largely depend on the filter to be applied. Of the two filters presented here, the first one takes 30s and the second one takes 2 min.

4 Discussion and ongoing research

The 3D modeling of nucleic acids (NA) requires context-specific analysis of NA conformations, beyond simple queries of pre-existing databases. The present manuscript describes protNAff, a novel tool to create customized databases of protein-bound NA 3D structures from the PDB, to select subsets of structures or sub-structures by any combination of filters, and/or to create fragment libraries of chosen characteristics.

Each output - database, subsets, fragments - can be useful independently: search for particular examples of complexes, statistical analyses on patterns, fragment-based modeling, etc. Our tool has been designed to be highly modular and adaptable to a very broad variety of possible usages, assuming a minimum of coding skills from the user. Thus, in contrast to a majority of existing databases and libraries, the primary audience of protNAff is bioinformaticians working on computational methods related to RNA structures. ProtNAff is freely available on GitHub at https://github.com/isaureCdB/protNAff. The GitHub repository contains the source code, user manual and installation instructions, as well as documented examples in the form of Jupyter notebooks. All figures and analyses in this study have been generated from these notebooks, giving an idea of the kind of biological insights that can be obtained. Starting from these examples, protNAff is meant to serve as a resource for bioinformaticians to create their own filtering and analysis scripts. We foresee that such scripts can eventually be leveraged into web services accessible to a broader audience of biologists. In order to demonstrate this possibility, we have written a prototype Google Colab web service. This web service runs in the browser without visible Python code or installation by the user.

Acknowledgements

References

Becquey, L., Angel, E., and Tahi, F. (2021). Rnanet: an automatically built dual-source dataset integrating homologous sequences and rna structures. Bioinformatics, **37**, 1218 – 1224.

Bhattacharya, D., Adhikari, B., Li, J., and Cheng, J. (2016). FRAGSION: ultra-fast protein fragment library generation by IOHMM sampling. Bioinformatics, **32**(13).

de Beauchene, I. C., de Vries, S., and Zacharias, M. (2016). Fragment-based modelling of single stranded RNA bound to RNA recognition motif containing proteins. Nucleic Acids Research, **44**(10), 4565–4580.

de Vries, S., Schindler, C., de BeauchÃªne, I. C., and Zacharias, M. (2015). A web interface for easy flexible protein-protein docking with ATTRACT. Biophysical journal, **3**, 462–465.

Dolinsky, T., Czodrowski, P., Li, H., Nielsen, J., Jensen, J., Klebe, G., and Baker, N. (2007). PDB2PQR: expanding and upgrading automated preparation of biomolecular structures for molecular simulations. Nucleic Acids Research, **35**, 522–525.

Fulle, S. and Gohlke, H. (2008). Analyzing the flexibility of rna structures by constraint counting. Biophysical journal, **94**, 4202–4219.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with NumPy. Nature, **585**(7825), 357–362.

Jones, S. (2016). ProteinâŁ“RNA interactions: structural biology and computational modeling techniques. Biophysical Reviews, **8**(4), 359–367.

Lu, X. and Olson, W. (2003). 3DNA: a software package for the analysis, rebuilding and visualization of threeâŁłdimensional nucleic acid structures. Nucleic Acids Research, **31**, 5108–5121.

Maris, C. and C. Dominguez, F. H.-T. A. (2005). The rna recognition motif, a plastic rna-binding platform to regulate post-transcriptional gene expression. The FEBS Journal, **272**, 2118–2131.

Masaki, S., Ikeda, S., Hata, A., Shiozawa, Y., Kon, A., Ogawa, S., Suzuki, K., Hakuno, F., Takahashi, S.-I., and Kataoka, N. (2019). Myelodysplastic syndrome-associated SRSF2 mutations cause splicing changes by altering binding motif sequences. Frontiers in genetics, **10**.

Mias-Lucquin, D. and de Beauchene, I. C. (2021). Conformational variability in proteins bound to single-stranded dna: A new benchmark for new docking perspectives. Proteins.

Moniot, A., de Vries, S., Ritchie, D., and de Beauchene, I. C. (2019). NAfragDB: a multi-purpose structural database of nucleic-acidâŁ“protein complexes for advanced users. In GGMM, page 21.

Moniot, A., de Beauchêne, I. C., and Guermeur, Y. (2022). Inferring Epsilon-nets of Finite Sets in a RKHS. Technical report, hal-03651323.

Richardson, K., Kirkpatrick, C., and Znosko, B. (2020). RNA CoSSMos 2.0: an improved searchable database of secondary structure motifs in rna three-dimensional structures. Database (Oxford).

Sagendorf, J. M., Markarian, N., Berman, H. M., and Rohs, R. (2020). DNAproDB: an expanded database and web-based tool for structural analysis of DNAâŁ“protein complexes. Nucleic Acids Research, **48**(1).

Zheng, G., Colasanti, A. V., Lu, X., and Olson, W. K. (2010). 3DNALandscapes: a database for exploring the conformational features of DNA. Nucleic Acids Research, **38**(1).