

METHODS AND APPROACHES

# ALLIN: A tool for annotation of a protein alignment combined with structural visualization

Alessandra Picollo<sup>1,2</sup>  and Michael Pusch<sup>1,2</sup> 

The physiological, functional, and structural properties of proteins and their pathogenic variants can be summarized using many tools. The information relating to a single protein is often spread among different sources requiring different programs for access. It is not always easy to select, simultaneously visualize, and compare specific properties of different proteins. On the other hand, comparing members of the same protein family could suggest conserved properties or highlight significant differences. We have thus developed a web interface, ALLIN (Annotation of sequence alignment and structural protein visualization) for the simultaneous visualization of multi-sequence protein alignments, including comments and annotations, and the related three-dimensional structures. This interface permits the inclusion of comments and coloring of residues in the alignment section, according to a user-defined color code, allowing a quick overview of specific properties. The interface does not require training or coding expertise, and the result is a unique “memo” web page that combines data from different sources, with the flexibility to highlight only the information of interest. The output provides an overview of the state of art of a protein family that is easily shared among researchers and new data can be conveniently added as it emerges. We believe the ALLIN tool can be useful for all scientists working on the structure–function analysis of proteins, in particular on those involved in human genetic diseases.

## Introduction

Most proteins belong to a protein family, or superfamily, whose members share a common 3-D architecture, similar functional properties, and significant sequence similarity. However, each member has a specific physiological role, specific biophysical properties, and often different pathological implications, in cases of genetic mutations. Structural and functional information obtained for one member can provide useful indications for the investigation of other proteins of the same family. The global comparison of these data can guide the identification of conserved regions or highlight structural and/or sequence differences. The task of keeping track and visualizing information related to a specific protein family collected through years of research in a single page is not easy. Sequence comparisons via alignments and structural differences are most often visualized in separate programs, making a systematic and quick assessment, for example, of the possible impact on patients of a novel genetic variant cumbersome. In addition, information on the functional and pathologic impact of missense variants is spread in different databases (for example, ClinVar, Uniprot, and Ensembl), adding further difficulty to combining available information in a straightforward and quickly accessible way. While there are already valid tools to customize 3-D structure protein visualization (for example, ICN3D, Jmol, Chimera, and RCSB

PDB) and powerful tools to generate protein sequence alignment with comments included (Jalview), none of these allows the combination of alignment, custom annotation, and structural visualization of a protein family. Here we present a user-friendly web interface “ALLIN” (Annotation of sequence alignment and structural protein visualization) for quickly creating an organized and “personalized” web page that could be considered as a “memo.” New information can be incrementally added to the page, which allows thus a quick but comprehensive overview of the state of art of the results related to a complete protein family: 3-D structures, sequence alignment, pathogenic variants, and alteration of functional properties or any other properties that the user is interested in.

No server is needed to create the final web page or to host the page. The open-source code is entirely client-side, written in plain HTML and JavaScript using the JavaScript library 3Dmol.js for 3-D visualization (Rego and Koes, 2015). Moreover, the generation of the input files is very simple and does not require training or coding skills and it can be easily adapted by the user.

The web interface is available at <https://github.com/mikpusch/ALLIN>, where also all necessary JavaScript files can be downloaded.

<sup>1</sup>Institute of Biophysics, National Research Council, Genova, Italy; <sup>2</sup>RAISE Ecosystem, Genova, Italy.

Correspondence to Michael Pusch: [michael.pusch@ibf.cnr.it](mailto:michael.pusch@ibf.cnr.it).

© 2025 Picollo and Pusch. This article is distributed under the terms as described at <https://rupress.org/pages/terms102024/>







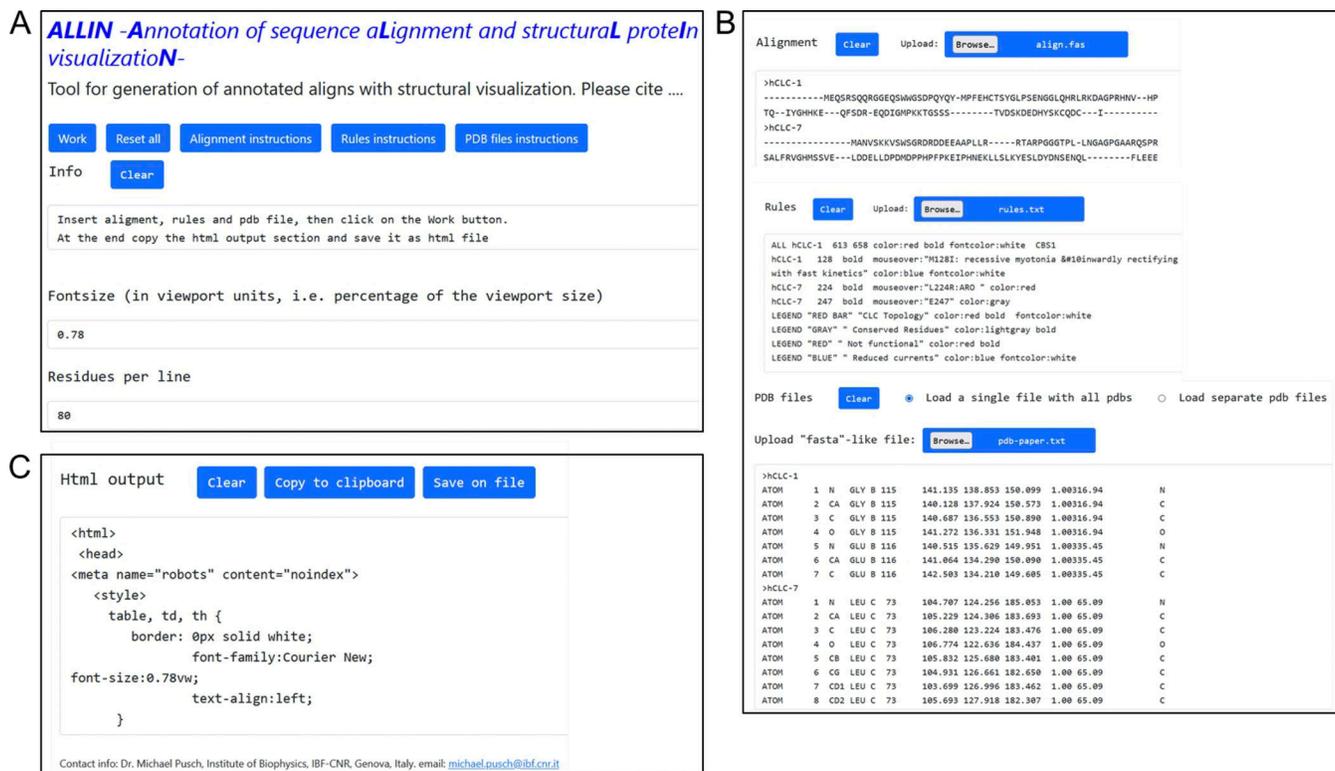


Figure 3. **Overview of the ALLIN web interface.** View of the three sections of the web interface. **(A)** Section 1 contains the info field and the two inputs that define the font size and the number of residues per line that will be shown in the multi-sequence alignment. At the top, five buttons are present: to start the generation of the HTML output (“work”), to clean all fields (“Reset all”), and three instruction buttons. **(B)** The three input fields with examples of a fasta alignment, a set of rules, and protein atomic coordinates. **(C)** Example of an HTML output script.

As explained in detail below, to generate the annotated alignment, `Align.html`, three pieces of input need to be provided:

- (1) An obligatory alignment, in fasta format, of the members of the protein family.
- (2) A nonobligatory set of annotations for selected (or all) residues.
- (3) For each member, a nonobligatory pdb file of the 3-D structure or a unique textfile containing the atomic coordinates of all structures of interest.

## Step-by-step protocol

The web interface is organized into five text fields: an info field providing help, error messages, and info on the outcome; three input fields (alignment, rules, and pdbs); and a final output field. A series of buttons allows for interaction with each field. After having populated the three input fields, clicking the “work” button starts the generation of the output html. Clicking the “Reset all” button resets the content of all fields.

In the info field at the top of the page (Fig. 3 A), messages describing the status of the running program are shown. For example, during execution, a message “working...” will appear. When the procedure is finished, the message “Finished! Now copy the text from the HTML field and save it as HTML” will appear. In addition, messages associated with specific errors

encountered during parsing the content of the input fields will be shown here. Finally, help text invoked by clicking one of the three instruction buttons is shown in the info field.

The “Fontsize” and “Residues per line” inputs allow defining the layout and the format of the sequence alignment in the final HTML file. “Fontsize” is in viewport units; in this way, the alignment will be screen-size scalable. The second box allows choosing how many residues per line will be shown on the final web page. Since the two output columns have fixed proportions, it is important to highlight that if the users prefers to increase the font size, they have to simultaneously reduce the number of residues per line. Each input box has a “Clear” button that allows cancelling the text contained in it and an “upload” button to load the input files and insert its content into the respective input field.

The generation of the final HTML web page requires the insertion of three types of information in the “Alignment,” “Rules,” and “PDB files” fields (Fig. 3 B).

### Step 1: Generation of the alignment file

The fasta multiple sequence alignment can be generated using any type of sequence alignment software (we used Clustal Omega). The resulting fasta file (\*.fas) has to be uploaded in this field. The format of the alignment file is given in Fig. 3 B. The names in the sequence identification code (for example, >hCLC-1 and >hCLC-7) have to be maintained identical in the

other two input fields (rules and pdb, Fig. 3 B). In the example align-CLC.fas file, we have included the sequence of the Torpedo tmCLC-0 and the sequences of the human CLCs (except CLC-Ka). All files are available on <https://github.com/mikpusch/ALLIN>. We suggest downloading the entire ALLIN folder to a local directory, to better consult the examples.

### Step 2: Generation of the rules file

The rules file contains the information used to add graphical visualization to the multi-protein alignment and to show residue-related information. Three different rule types are defined (see also Fig. 3 B for specific examples):

- (1) Rules used to mark segments of the structure, for example, transmembrane helices. These rules start with the keyword ALL (see below for details).
- (2) Rules applying to a specific residue of a specific protein. These rules start with the fasta protein identifier (see below for details).
- (3) Rules to illustrate the use of color codes and font styles. These rules start with the keyword LEGEND (see below for details).

The order of the rules in the file is irrelevant. However, for each residue, only one rule can be applied. If more than one rule is present in the rules field for a specific residue, only the first one will be used.

The first type allows inserting a colored bar, with a text repetition, under a region of multi-sequence alignment. An example is the following string:

```
ALL hCLC-1 613 658 color:red bold fontcolor:white CBS1
```

In this example, the text “CBS1,” highlighted in red and written in white bold, is repeatedly shown below the alignment in the stretch between residues 613 and 658 (using the numbering corresponding to the hCLC-1 sequence) to indicate that the selected region of residues forms a structural unit called CBS1.

To color and insert information related to a specific residue of a specific protein, a string similar to that provided in the following example has to be provided:

```
hCLC-1 128 bold mouseover:“M128I: recessive myotonia inwardly rectifying with fast kinetics” color:blue fontcolor:white
```

The first fasta tag “hCLC-1” defines the respective sequence. The number defines the corresponding residue of interest in the sequence. The tag “bold” allows for choosing the text style (italic is an additional option). With the tag “mouseover:”, it is possible to insert a multiline string that will be shown in a small box when the mouse hovers above the residue. The following combination of symbols “&#10” inserts a line break. The last two tags define the font color and the background color of the selected residue. It is important that no space has to be inserted after the colon in *mouseover:’...’m*, *fontcolor:white*, and *color:blue*.

Finally, to create a legend related to the significance of colors and font styles in the sequence alignment, a third type of rule can be used. These start with the keyword LEGEND, followed by the defining characteristic (for example, the color) and its corresponding description. The text color and font styles are then

added exactly as in the other two rules, as in the following example, such that the text “BLUE” is shown according to the font specifications (see Fig. 1):

```
LEGEND “BLUE” “Altered ion transport properties” color:blue fontcolor:white bold
```

### Step 3: Insertion of PDB files

For each protein of the sequence alignment, it is possible, but not necessary, to insert a 3-D structure, solved experimentally or generated as a homology model.

The user can choose to upload a single text file containing the atomic coordinates of all proteins with a fasta definition line (i.e., >hCLC-1) separating each protein or upload each pdb file one by one. It is important to stress that the fasta identifier in the text file containing the pdb files and that of the sequence alignment must be identical. Pdb files usually contain a lot of information, but here only the atomic coordinates are necessary, as shown in Fig. 3 B, bottom. If for the same entry of the alignment (e.g., >hCLC-1), more than one pdb entry is present in the pdb field, only the last pdb entry will be used. For the example of CLC proteins, the folder CLC contains individual pdb files and a text file (final\_pdb.txt) with all CLC structures.

### Step 4: Generation of the HTML output

Once all input fields have been populated, clicking the work button at the top of the page, the final HTML script will be generated and will appear in the HTML output field. This script contains the web page style instructions and few lines of source code to embed the JavaScript libraries (3Dmol-min.js, jquery.min.js, and My3dMol-JavaScript.js) necessary to create and manipulate the 3-D molecular visualizations.

The content can be saved on a file clicking the “Save on file” button, which should be stored in the same folder as the JavaScript files.

### Data availability

The GitHub repository <https://github.com/mikpusch/ALLIN> includes two example folders: the folder “CLC” contains the files used for the “case study” discussed in the paper, including a sub-folder with all pdb files of structures and models of CLCs. To familiarize with the ALLIN tool, a second folder “KCNQ” is provided. This example includes the three text files, align-Kcnq.txt, rules-Kcnq.txt, and pdb-Kcnq.txt, containing the information that has to be loaded into the three input fields for alignment, rules, and pdb files, respectively, to generate the final Kcnq.html file.

### Acknowledgments

Joseph A. Mindell served as editor.

This research was funded by the European Union—NextGenerationEU (Missione 4 Componente 2, “Dalla ricerca all’impresa,” Innovation Ecosystem RAISE “Robotics and AI for Socio-economic Empowerment,” ECS00000035), by D34Health (grant CUP B83C22006120001), and by the Italian Ministry of Foreign Affairs (grant CUP B53C23008450001).

Author contributions: A. Picollo: conceptualization, funding acquisition, Software, validation, visualization, and writing—original draft, review, and editing. M. Pusch: conceptualization, data curation, formal analysis, funding acquisition, investigation, methodology, project administration, resources, software, validation, visualization, and writing—original draft, review, and editing.

Disclosures: The authors declare no competing interests exist.

Submitted: 8 July 2024

Revised: 23 September 2024

Revised: 31 October 2024

Accepted: 10 December 2024

## References

- Depienne, C., M. Bugiani, C. Dupuits, D. Galanaud, V. Toutou, N. Postma, C. van Berkel, E. Polder, E. Tollard, F. Darios, et al. 2013. Brain white matter oedema due to CLC-2 chloride channel deficiency: An observational analytical study. *Lancet Neurol.* 12:659–668. [https://doi.org/10.1016/S1474-4422\(13\)70053-X](https://doi.org/10.1016/S1474-4422(13)70053-X)
- Duncan, A.R., M.M. Polovitskaya, H. Gaitán-Peñas, S. Bertelli, G.E. VanNoy, P.E. Grant, A. O'Donnell-Luria, Z. Valivullah, A.K. Lovgren, E.M. England, et al. 2021. Unique variants in CLCN3, encoding an endosomal anion/proton exchanger, underlie a spectrum of neurodevelopmental disorders. *Am. J. Hum. Genet.* 108:1450–1465. <https://doi.org/10.1016/j.ajhg.2021.06.003>
- Hu, H., S.A. Haas, J. Chelly, H. Van Esch, M. Raynaud, A.P. de Brouwer, S. Weinert, G. Froyen, S.G. Frints, F. Laumonier, et al. 2016. X-exome sequencing of 405 unresolved families identifies seven novel intellectual disability genes. *Mol. Psychiatry.* 21:133–148. <https://doi.org/10.1038/mp.2014.193>
- Jentsch, T.J., and M. Pusch. 2018. CLC chloride channels and transporters: Structure, function, physiology, and disease. *Physiol. Rev.* 98:1493–1590. <https://doi.org/10.1152/physrev.00047.2017>
- Jentsch, T.J., K. Steinmeyer, and G. Schwarz. 1990. Primary structure of Torpedo marmorata chloride channel isolated by expression cloning in *Xenopus* oocytes. *Nature.* 348:510–514. <https://doi.org/10.1038/348510a0>
- Kornak, U., D. Kasper, M.R. Bösl, E. Kaiser, M. Schweizer, A. Schulz, W. Friedrich, G. Dellling, and T.J. Jentsch. 2001. Loss of the CLC-7 chloride channel leads to osteopetrosis in mice and man. *Cell.* 104:205–215. [https://doi.org/10.1016/S0092-8674\(01\)00206-9](https://doi.org/10.1016/S0092-8674(01)00206-9)
- Lorenz, C., C. Meyer-Kleine, K. Steinmeyer, M.C. Koch, and T.J. Jentsch. 1994. Genomic organization of the human muscle chloride channel CLC-1 and analysis of novel mutations leading to Becker-type myotonia. *Hum. Mol. Genet.* 3:941–946. <https://doi.org/10.1093/hmg/3.6.941>
- Ma, T., L. Wang, A. Chai, C. Liu, W. Cui, S. Yuan, S. Wing Ngor Au, L. Sun, X. Zhang, Z. Zhang, et al. 2023. Cryo-EM structures of CLC-2 chloride channel reveal the blocking mechanism of its specific inhibitor AK-42. *Nat. Commun.* 14:3424. <https://doi.org/10.1038/s41467-023-39218-6>
- Miller, C., and M.M. White. 1980. A voltage-dependent chloride conductance channel from Torpedo electroplax membrane. *Ann. N. Y. Acad. Sci.* 341:534–551. <https://doi.org/10.1111/j.1749-6632.1980.tb47197.x>
- Park, E., E.B. Campbell, and R. MacKinnon. 2017. Structure of a CLC chloride ion channel by cryo-electron microscopy. *Nature.* 541:500–505. <https://doi.org/10.1038/nature20812>
- Park, E., and R. MacKinnon. 2018. Structure of the CLC-1 chloride channel from *Homo sapiens*. *Elife.* 7:e36629. <https://doi.org/10.7554/eLife.36629>
- Rego, N., and D. Koes. 2015. 3Dmol.js: Molecular visualization with WebGL. *Bioinformatics.* 31:1322–1324. <https://doi.org/10.1093/bioinformatics/btu829>
- Schrecker, M., J. Korobenko, and R.K. Hite. 2020. Cryo-EM structure of the lysosomal chloride-proton exchanger CLC-7 in complex with OSTMI. *Elife.* 9:e59555. <https://doi.org/10.7554/eLife.59555>
- Wan, Y., S. Guo, W. Zhen, L. Xu, X. Chen, F. Liu, Y. Shen, S. Liu, L. Hu, X. Wang, et al. 2024. Structural basis of adenine nucleotides regulation and neurodegenerative pathology in CLC-3 exchanger. *Nat. Commun.* 15:6654. <https://doi.org/10.1038/s41467-024-50975-w>
- Wang, K., S.S. Preisler, L. Zhang, Y. Cui, J.W. Missel, C. Grønberg, K. Gotfryd, E. Lindahl, M. Andersson, K. Calloe, et al. 2019. Structure of the human CLC-1 chloride channel. *PLoS Biol.* 17:e3000218. <https://doi.org/10.1371/journal.pbio.3000218>
- Zhang, B., S. Zhang, M.M. Polovitskaya, J. Yi, B. Ye, R. Li, X. Huang, J. Yin, S. Neuens, T. Balfroid, et al. 2023. Molecular basis of CLC-6 function and its impairment in human disease. *Sci. Adv.* 9:eadg4479. <https://doi.org/10.1126/sciadv.adg4479>
- Zhang, S., Y. Liu, B. Zhang, J. Zhou, T. Li, Z. Liu, Y. Li, and M. Yang. 2020. Molecular insights into the human CLC-7/Ostml transporter. *Sci. Adv.* 6:eabb4747. <https://doi.org/10.1126/sciadv.abb4747>