



HAL
open science

Extracting Dynamical Correlations and Identifying Key Residues for Allosteric Communication in Proteins by **correlationplus**

Mustafa Tekpinar, Bertrand Neron, Marc Delarue

► **To cite this version:**

Mustafa Tekpinar, Bertrand Neron, Marc Delarue. Extracting Dynamical Correlations and Identifying Key Residues for Allosteric Communication in Proteins by **correlationplus**. *Journal of Chemical Information and Modeling*, 2021, 61 (10), pp.4832-4838. 10.1021/acs.jcim.1c00742 . pasteur-03864465

HAL Id: pasteur-03864465

<https://hal-pasteur.archives-ouvertes.fr/pasteur-03864465>

Submitted on 21 Nov 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Extracting Dynamical Correlations and Identifying Key Residues for Allosteric Communication in Proteins by *correlationplus*

Mustafa Tekpinar^{1*}, *Bertrand Neron*², *Marc Delarue*^{1*}

¹Unit of Architecture and Dynamics of Biological Macromolecules, Pasteur Institute, UMR 3528
CNRS, 25 Rue du Dr. Roux, 75015 Paris, France.

²Bioinformatics and Biostatistics Hub, Computational Biology Department, 28 Rue du Dr. Roux,
75015 Paris, France.

KEYWORDS. Allostery, coevolution, coupling, dynamical cross-correlation, elastic network
model, linear mutual information, molecular dynamics simulations.

ABSTRACT. Extracting dynamical pairwise correlations and identifying key residues from large molecular dynamics trajectories or normal mode analysis of coarse-grained models are important for explaining various processes like ligand binding, mutational effects, and long-distance interactions. Efficient and flexible tools to perform this task can provide new insights about residues involved in allosteric regulation and protein function. In addition, combining and comparing dynamical coupling information with sequence coevolution data can help to understand better protein function. To this aim, we developed a Python package called *correlationplus* to calculate, visualize and analyze pairwise correlations. In this way, the package aids to identify key residues and interactions in proteins. The source code of *correlationplus* is available under LGPL version 3 at <https://github.com/tekpinar/correlationplus>. The current version of the package (0.2.0) can be installed with common installation methods like conda or pip in addition to source code installation. Moreover, docker images are also available for usage of the code without installation.

INTRODUCTION.

Proteins are macromolecular machines performing essential functions in the cell. Protein function is intertwined to protein dynamics, structure and sequence. Because the amount of dynamical, structural and genomic data on proteins is increasing very rapidly, integrating all this information and inferring meaningful relations relevant to protein function from this data necessitates new analysis techniques.

In particular, extracting information from dynamical data and connecting this data to protein function is a major challenge. Protein dynamics can be studied with various experimental and computational methods. Molecular dynamics (MD) simulations and normal mode analysis of coarse-grained models such as the elastic network models (ENMs) are among the most widely used computational methods to study the dynamical properties of proteins. These two methods produce large amounts of data to be analyzed. Extracting pairwise couplings from this data and identifying key residues playing a role in communication between different parts of a protein is essential because it has been shown that the computed dynamical couplings agree with different experimental data for various proteins¹⁻⁵. Computing pairwise dynamical couplings requires fast, flexible and accessible tools. As a result, developing such tools is a common theme in structural bioinformatics. Here, we developed a Python package called *correlationplus* to calculate, visualize and analyze pairwise correlations and identify key residues in proteins.

Recently, sequence coevolution data analysis combined with machine learning methods resulted in impressive breakthroughs in computational protein structure determination^{6,7}. These studies, along with many others, indicate that protein sequences may contain information about protein structure, function, dynamics and potentially allostery. Due to this reason, *correlationplus* includes some functionality to process and analyze sequence coevolution data or any other

coupling data given in a full matrix format. In this way, we aim to build a bridge between protein dynamics and sequences to infer meaningful relations relevant to protein function and allostery.

THEORETICAL BACKGROUND

Coarse-grained Elastic Network Models (CG-ENMs)

There are numerous forms of elastic network models with different levels of atomic details and force fields. Gaussian network model (GNM)⁸, anisotropic network model (ANM)⁹, parameter-free ANM¹⁰, rotation-translation blocks (RTB)¹¹, sequence and distance dependent ENM¹² are some commonly used models. Even though all-atoms of a system can be considered for normal mode analysis, selecting some atoms to represent an amino acid, namely coarse-graining, reduces computational work significantly. Due to this advantage, we used only coarse-grained models in this work. *Correlationplus* allows the users to compute normal modes based on coarse-grained (CG) GNM and ANM, which are explained further below.

Gaussian Network Model (GNM)

In CG-GNM, all C α atoms (N particles) of a protein within a cutoff radius ($R_c=10$ Å by default) are connected with springs of uniform stiffness ($\gamma=1.0$) and they interact through the potential given below^{8, 13}

$$V_{GNM} = \frac{\gamma}{2} \left[\sum_{i,j}^N \theta(R_c - R_{ij}) (\mathbf{R}_{ij} - \mathbf{R}_{ij}^0)^2 \right] = \frac{\gamma}{2} \left[\sum_{i,j}^N (\Delta \mathbf{R}_i \Gamma_{ij} \Delta \mathbf{R}_j) \right] \quad (1)$$

$\theta(R_c - R_{ij})$ denotes Heaviside step function and it ensures that only interactions within a cutoff radius (R_c) are considered. \mathbf{R}_{ij} is instantaneous distance vector between C α atoms of residue i and

j, while \mathbf{R}_{ij}^0 is the equilibrium distance vector between the same atoms. $\Delta \mathbf{R}_j = \mathbf{R}_j - \mathbf{R}_j^0$ is distance difference vector of C α atom of residue j. \mathbf{R}_j is the instantaneous position vector for C α atom of residue j at any moment while \mathbf{R}_j^0 is equilibrium position vector for C α atom of residue j. Γ_{ij} is ij^{th} element of Kirchoff matrix Γ and this element is zero for interactions beyond the cutoff radius, it is -1 for all distinct i and j elements. N-1 non-zero eigenvectors of the Kirchoff matrix constitute the normal modes of CG-GNM, where N is number of C α atoms.

Anisotropic Network Model (ANM)

In CG-ANM, all C α atoms of a protein within a certain cutoff radius ($R_c=15 \text{ \AA}$ by default) interact with each other through a simple Hookean potential⁹:

$$V_{ANM} = \frac{\gamma}{2} \left[\sum_{i,j}^N \theta(R_c - R_{ij}) (R_{ij} - R_{ij}^0)^2 \right] \quad (2)$$

In this potential, γ has the same definition and default value as in GNM. R_{ij}^0 denotes the *equilibrium distance* between C α atoms of residue i and j, while R_{ij} is the *instantaneous distance* between the same pair of C α atoms. Using this potential, one can build a hessian matrix and find its eigenvectors to obtain 3N-6 normal modes of a protein, where N is the number of C α atoms . This model has been successful in explaining experimentally observed conformational changes of proteins as well as fluctuations around native conformations and coupled dynamics of proteins^{9, 14, 15}.

Linear Correlations from Dynamics

There are two widely used linear correlation metrics that can be obtained from dynamics data: dynamical cross-correlations (DCC) and linear mutual information (LMI) of protein fluctuations.

These computed linear correlations can provide insights about dynamical couplings in proteins. Previous studies showed that the information obtained from the dynamical correlations are supported by various experimental methods^{1, 16}. For example, Chesca NMR correlations² are in qualitative agreement with both DCC and LMI results for Chemotactic Y protein¹⁷, which is an allosteric protein (See **Figure S1**). In addition, the correlations can be used for automated domain classification and dynamical community analyses¹⁸⁻²⁰. As a result, the dynamical correlations can be useful by themselves or they can be processed further by graph theoretical approaches to extract more information. Theoretical background on how to calculate DCC and LMI from both MD trajectories or normal modes obtained from CG-ENMs is introduced below.

Dynamical Cross-Correlation (DCC)

One can calculate DCC between residues i and j from a set of models. The source of these models can be experimental methods like NMR or theoretical methods such as MD trajectories or normal modes. DCC can be defined as follows:

$$DCC_{ij} = \langle \Delta \mathbf{R}_i \cdot \Delta \mathbf{R}_j \rangle \quad (3)$$

Here, $\Delta \mathbf{R}_i = \mathbf{R}_i - \langle \mathbf{R}_i \rangle$ is the positional difference vector of atom i with respect to its average

position after an overall fitting procedure has been conducted. $\langle \rangle$ indicates that an averaging over selected models is performed.

In the case of normal modes, DCC can be obtained with Equation (4).

$$DCC_{ij} = \frac{3k_B T}{\gamma} \sum_{l=1}^L \frac{\mu_l^i \mu_l^j}{\lambda_l} \quad (4)$$

Here, k_B denotes Boltzmann constant and T is temperature. λ_l denotes eigenvalue of mode l and μ_l is the corresponding eigenvector. It is convenient to normalize DCCs as follows:

$$nDCC_{ij} = \frac{DCC_{ij}}{[DCC_{ii} DCC_{jj}]^{1/2}} \quad (5)$$

nDCC ranges between [-1, 1]. While 0 indicates no correlations, 1 (-1) means complete correlation (anti-correlation). If we are only interested in the existence of the correlations between residue pairs rather than the direction of the correlations, only the absolute values of nDCCs can be used. In this case, all values will be between [0, 1].

Linear Mutual Information (LMI)

Despite its widespread usage, nDCC has some problems. If two atoms move in perpendicular directions simultaneously, their correlation cannot be measured with nDCC due to the dot product in Equation 3. **Instead**, LMI is a parameter that does not suffer from angular dependency as encountered in DCC²¹. To calculate LMI between residues i and j , one has to calculate $C_i = \langle \mathbf{x}_i^T \mathbf{x}_i \rangle$ and $C_{ij} = \langle (\mathbf{x}_i, \mathbf{x}_j)^T (\mathbf{x}_i, \mathbf{x}_j) \rangle$ where $\mathbf{x}_i = \mathbf{R}_i - \langle \mathbf{R}_i \rangle$. Here, \mathbf{R}_i denotes position vector of atom i .

Inserting C_i , C_j and C_{ij} into the following equation

$$LMI_{ij} = \frac{1}{2} [\ln (\det C_i) + \ln (\det C_j) - \ln (\det C_{ij})] \quad (6)$$

gives us the pairwise LMI. When the values are normalized, 0 value of LMI means that there is no correlation, while 1 means they are completely correlated.

DESCRIPTION AND FEATURES

Correlationplus is a Python 3 package that can be executed on Linux, Mac OS and Windows operating systems. Even though it can be installed from the source code given at <https://github.com/tekpinar/correlationplus>, pip/conda based installation procedures were implemented to facilitate the installation. *Correlationplus* utilizes several high-quality Python packages like Numpy²², Matplotlib²³, Prody^{24, 25}, NetworkX²⁶ and MDAnalysis^{27, 28}. However, the end-user does not have to preinstall these packages because the dependencies are handled automatically by pip or conda. Due to multiple operating system support and automated installation procedures, *correlationplus* aims to reach a broad audience of computational and experimental structural biology community.

Correlationplus contains four major modules that can be used both via command-line script interfaces as well as interactively. The modules are named ‘calculate’, ‘visualize’, ‘centralityAnalysis’ and ‘pathsAnalysis’. To facilitate the usage of the modules, there is a script interface for each module. The names of the scripts are as follows: ‘calculate’, ‘visualize’, ‘analyze’ and ‘paths’. After installation of the package, each script can be invoked from the terminal as follows: ‘correlationplus calculate -h’ etc. Each script will be introduced in the following subsections.

Calculate Script

The ‘calculate’ script of *correlationplus* can compute DCCs²⁹ and LMIs between pairs of residues²¹ (**Figure 1 a**). While only a PDB file is sufficient for CG-ENM-based computations, MD coordinate trajectories need to be provided to extract dynamical correlations from MD simulations. ENM-based approach can yield dynamical information quickly albeit with limited atomic details. On the other hand, MD-based data can provide atomic details such as the influence of ligand binding or mutational effects, but at a higher computational cost. Users can select the method according to their needs and resources.

Prody package²⁵ is utilized for the calculation of normal modes according to the gaussian network model (GNM) or the anisotropic network model (ANM). GNM can be more efficient for large protein complexes. There are recommended values for number of normal modes (100) and the cutoff distance (10 Å for GNM and 15 Å for ANM)¹⁵, but these values can be tuned by the user via command line arguments. Even though only two CG-ENMs (GNM and ANM) are used to compute normal modes in *correlationplus*, there is not any restriction on the usage of the other CG-ENMs. As a result, they can also be used directly (by providing the normal modes as trajectory files) or indirectly (by providing the DCC or LMI matrices).

Correlationplus can handle common MD trajectory formats like dcd, xtc and trr. Thanks to the command-line arguments implemented, only a certain part of a trajectory can also be selected in the calculations. Trajectory handling support is provided by the highly efficient MDAnalysis Python package^{27, 28}. In fact, even models obtained from NMR experiments or models that are not based on MD simulations³⁰ can be used to compute DCC or LMI, if converted to an appropriate trajectory format. This ability of *correlationplus* to handle different formats and to process data from various sources gives it a high flexibility and increases its general applicability.

Typical output of this script is a full matrix of the selected correlation quantity. This matrix can be used -along with a protein structure in PDB format- in ‘visualize’, ‘analyze’ and ‘paths’ scripts, which are described in the following subsections.

Visualize Script

The ‘visualize’ script of *correlationplus* is integrated with the commonly used protein visualization software VMD³¹ and PyMOL³² to produce high quality 3D visualizations. This script produces tcl/pml scripts that can be called with VMD/PyMOL to project correlations onto protein structures. Each pairwise correlation is depicted as a blue cylinder whose radius is proportional to the strength of the correlation. In its current implementation, normalized dynamical correlation values higher than $|0.75|$ are projected onto the protein structure. However, this default value can be changed easily with script arguments. If only the correlations between distant residues are investigated, a distance filter can also be applied. In this way, only the correlations related to long-distance interactions can be projected on the structure.

Furthermore, the ‘visualize’ script produces high quality 2D figures of correlation maps. If the protein contains multiple chains, the script can produce inter-chain and intra-chain correlations separately (**Figure 1 b**). In this way, subunit and domain cooperativity mechanisms can be studied. Also, this script produces a 2D plot of the distribution of the correlation quantity with respect to inter-residue distances. This information can be useful to evaluate the potential of a given correlation metric to identify long-range allosteric interactions.

Even though the ‘calculate’ script can only compute dynamical correlations, the ‘visualize’ script is not limited to these quantities. It can also be used to visualize various sequence coevolution/conservation matrices such as Pearson (pair) correlation coefficient, mutual

information and joint Shannon entropy obtained from multiple sequence alignments. Particularly, the output of CoeViz server can be used with the ‘visualize’ script with minimal effort³³. Moreover, the ‘visualize’ script can parse coevolution data from EVcouplings server (<https://evcouplings.org/>)³⁴. In addition to data from these two servers, the user can visualize any other coupling data given in a full matrix format by specifying the data type as ‘generic’. This gives the end-user a significant flexibility to visualize couplings in proteins. In this way, dynamical correlation data can be combined with sequence information obtained by various approaches to extract key residues playing different dynamical or structural roles.

Analyze Script

Combining results of MD simulations or normal modes of CG-ENMs with graph theoretical approaches can yield new insights about protein dynamics and their role in allostery. tRNA:protein complexes, imidazole glycerol phosphate synthase and CRISPR–Cas9 HNH Nuclease are some examples showing the potential of this type of analyses with experimental support³⁵⁻³⁸. Dynamical communities and various dynamics-based centrality metrics obtained from graph theoretical approaches revealed key residues playing a role in signal transmission and allosteric regulation³⁹⁻⁴¹. Due to these reasons, we implemented a special script to extract graph theoretical features such as degree⁴², betweenness⁴², closeness⁴², current flow betweenness⁴³, current flow closeness, eigenvector centrality⁴⁴ and community analyses⁴⁵ (**Figure 1 c**). In this script, a graph is constructed by considering C α atoms of the protein as nodes of the graph. If the residue correlations are higher than a value-threshold (0.3 for nDCC and LMI) and the inter-node distance is lower than a distance-threshold (7 Å), an edge is constructed between nodes *i* and *j*. Since absolute values of the normalized pairwise dynamical correlations are between [0, 1], the weight of each edge can be chosen as $-\log(|C_{ij}|)$ for a dynamics-based network, where C_{ij} is the relevant correlation quantity

between residues i and j . In this way, edge weights remain always positive. On the other hand, weight of each edge is $1/|C_{ij}|$ for a pairwise correlation value obtained from sequence coevolution metrics and or any other generic data to avoid negative weights for correlation/coupling values greater than 1. The weight is bidirectional in all cases; therefore, the network is an undirected network. The initial threshold values are the ones that are recommended in previous studies^{35, 46}, but users can modify them according to their needs.

We used the well-established NetworkX Python package to calculate graph theoretical quantities²⁶. After the calculations, all centralities are added to the B-factor column of a newly-created PDB file that can be visualized with any protein viewer by coloring $C\alpha$ atoms according to their B-factors. Previous studies^{47, 48} showed that active site residues, functional sites and key residues at inter-subunit interfaces can be identified with this kind of analyses (as an example, see the quantity called eigenvector centrality projected on SARS-CoV-2 main protease in **Figure 1 c**).

Paths Script

We can calculate and visualize optimal and suboptimal communication paths between a source residue and a target residue with the aid of 'paths' script. Only chain ID and a residue ID (as given in the PDB file) is needed both for the source and the target residues (For example, -b A41 -e B41 indicates that source residue is residue 41 of chain A and the target residue is residue 41 of chain B). The paths are weighted as explained in the previous subsection and they are calculated according to the algorithm described by Jin Y. Yen⁴⁹. In addition to the path lengths, tcl and pml scripts helps the end-user visualize these paths quickly with VMD and PyMOL. An example output for the suboptimal communication path between active sites of SARS-CoV-2 main protease is given in **Figure 1 d**.

DISCUSSION AND CONCLUSION

Here we discuss the main differences between currently available software and *correlationplus*. DCC calculation is implemented in several packages such as Carma⁵⁰, Prody^{24, 25} and Bio3D⁵¹. Prody is an essential component in the ‘calculate’ script of *correlationplus* for obtaining normal mode-based DCCs but it cannot be used to calculate LMI. In addition, visualization and dissection of the correlations as well as network analysis capabilities of *correlationplus* are features that make it significantly distinct from Carma and Prody packages. Bio3D⁵¹, which is an R package, has some functionalities similar to *correlationplus*. However, its trajectory handling (only dcd files) capabilities are more limited compared to *correlationplus*. MDAnalysis^{27, 28} also does not have these capabilities, even though it has some excellent features for trajectory handling and analysis. In terms of dynamical network analysis, WISP⁵² is an important software to note. It is integrated to VMD to provide visualization of long-range interactions. However, difficulties in its installation and its non-modular architecture fade its appeal for conducting dynamical network analysis with it. Another software with some dynamical network analysis capabilities is xPyder plugin⁵³, which is integrated to PyMOL. Even though its graphical user interface is very advantageous, *correlationplus* is more flexible since it allows the end-user to use both application programming interface and script interface. In addition, *correlationplus* allows calculation of more graph centralities than xPyder. A recent dynamical network analysis software that uses only generalized mutual information is also worthy of mentioning. This Python package is named *dynetan* and it can only calculate generalized mutual information quite efficiently. To summarize, even though there are several packages that has partially overlapping functionalities with our software, *correlationplus* differs from the other packages in terms of its design: It aims to integrate correlation/coupling data of different origins and extract long-distance interactions from it.

Correlationplus is a fast and easy-to-use tool to get information about pairwise correlations and to identify key functional components in a protein structure. For a protein with 1500 residues, the full workflow from calculation to all analyses of correlations can be completed within an hour on a typical workstation. In addition, it is possible to analyze and combine coevolution matrices with the dynamical information. In conclusion, *correlationplus* is an efficient tool to produce, visualize and analyze dynamical information rapidly and combine it with different information sources to understand key residues and pairwise interactions playing allosteric and functional roles in proteins.

DATA AND SOFTWARE AVAILABILITY

The *correlationplus* is open source and its source code is available under LGPL version 3 at <https://github.com/tekpinar/correlationplus>. **Figure 1** of this paper was obtained from the publicly available data in the *examples* folder of this repository. An example Bash script showing basic usage of *correlationplus* is also provided in the *examples* folder. The current version of the *correlationplus* package (0.2.0) can be installed with common installation methods like conda or pip in addition to the source code installation. Moreover, docker images are also available for usage of the code without installation.

FIGURES.

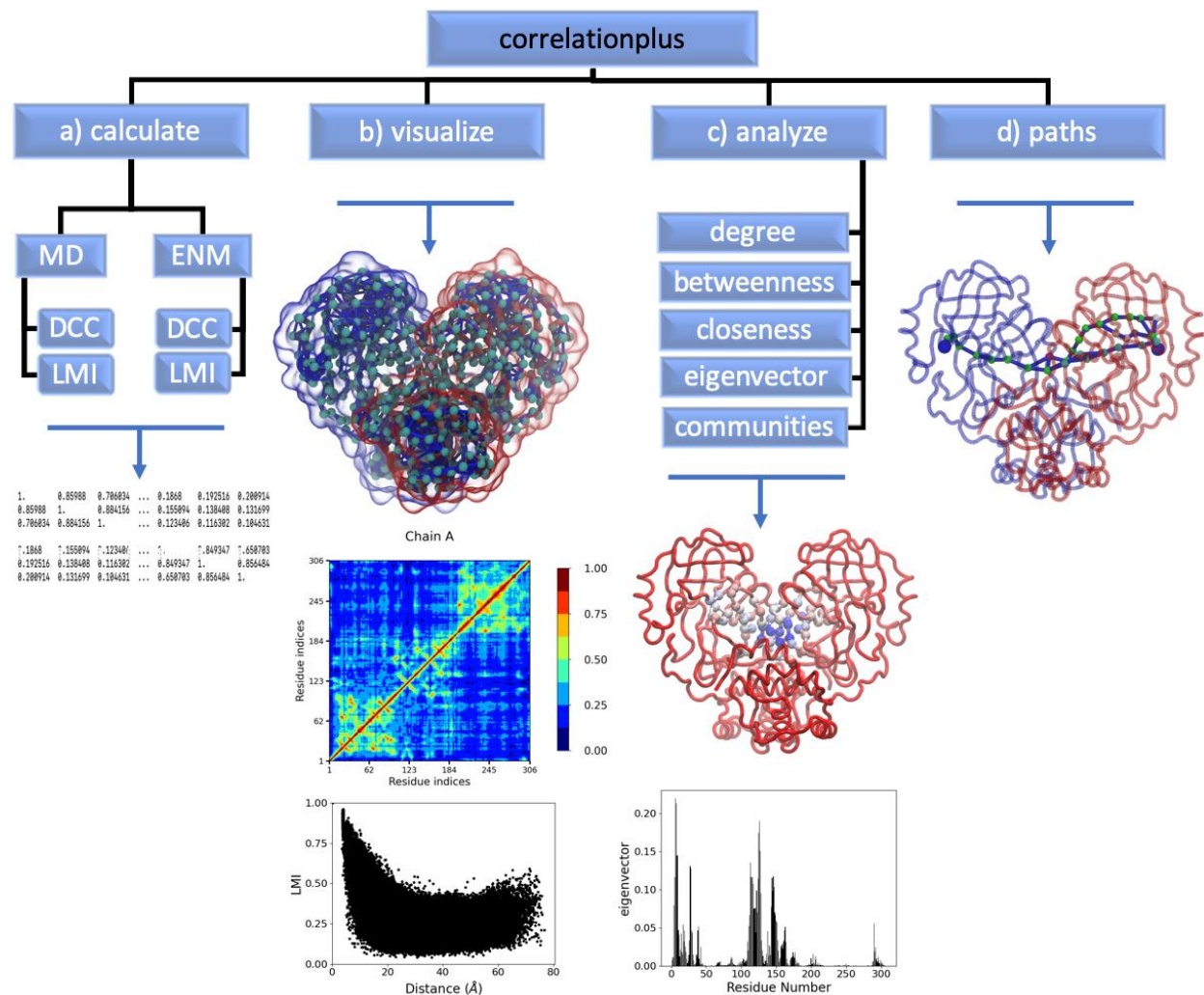


Figure 1. A summary of functionalities of *correlationplus*. a) calculate script b) visualize script c) analyze script output with an example showing the eigenvector centrality results for SARS-CoV-2 main protease⁴⁸. d) paths script output showing suboptimal allosteric communication paths between two active site residues of SARS-CoV-2 main protease.

ASSOCIATED CONTENT

Supporting Information.

The following file is available free of charge.

Supporting Information contains 1 figure (**Figure S1**) and an appendix (file type: PDF).

Figures

Figure S1. Absolute values of Chesca NMR correlations (upper triangle) vs computed correlations (lower triangle) for Chemotactic Y protein (PDB ID: 1f4v¹⁷ Chain A). **a)** Absolute values of Chesca NMR vs. absolute values of nDCC. **b)** Absolute values of Chesca NMR vs. normalized LMI. Three out of the four major peaks in LMI and nDCC matrices (red and dark red regions labeled with Roman numerals from I to IV) have corresponding high correlations in the experimental data.

AUTHOR INFORMATION

Corresponding Authors

Mustafa Tekpinar, Email: tekpinar@buffalo.edu; Marc Delarue, Email: delarue@pasteur.fr.

Phone: +33 01 40 61 31 11

Author Contributions

All authors read and approved the final version of the manuscript.

Notes

The authors declare no competing interests.

Erratum

A Python script showing versatility of the correlationplus application programming interface compared to the traditional webservers that perform similar tasks is provided in the correlationplus github repository:

https://github.com/tekpinar/correlationplus/blob/master/examples/example_api_usage.py

After the proofs of this article were corrected, we were made aware of the existence of the paper by Sheik Amamuddy et. al⁵⁴. This work developed tools similar to ours, but both were developed completely independently.

ACKNOWLEDGMENT

MT thanks to Programme PAUSE of College de France for their support.

ABBREVIATIONS

Anisotropic network model: ANM, Coarse-grained: CG, Dynamical Cross-Correlations: DCC,
Elastic Network Model: ENM, Gaussian network model: GNM, Linear Mutual Information: LMI.

REFERENCES

1. Kormos, B. L.; Baranger, A. M.; Beveridge, D. L., Do collective atomic fluctuations account for cooperative effects? Molecular dynamics studies of the U1A-RNA complex. *J Am Chem Soc* **2006**, *128* (28), 8992-3.
2. Selvaratnam, R.; Chowdhury, S.; VanSchouwen, B.; Melacini, G., Mapping allostery through the covariance analysis of NMR chemical shifts. *Proc. Natl. Acad. Sci. U. S. A.* **2011**, *108* (15), 6133-6138.
3. Akimoto, M.; Selvaratnam, R.; McNicholl, E. T.; Verma, G.; Taylor, S. S.; Melacini, G., Signaling through dynamic linkers as revealed by PKA. *Proc. Natl. Acad. Sci. U. S. A.* **2013**, *110* (35), 14231-14236.
4. Wang, J.; Jain, A.; McDonald, L. R.; Gambogi, C.; Lee, A. L.; Dokholyan, N. V., Mapping allosteric communications within individual proteins. *Nat Commun* **2020**, *11* (1), 3862.
5. Yildirim, A.; Tekpinar, M.; Wassenaar, T. A., Competing Roles of Ca(2+) and Nonmuscle Myosin IIA on the Dynamics of the Metastasis-Associated Protein S100A4. *J Phys Chem B* **2021**.
6. Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Zidek, A.; Potapenko, A.; Bridgland, A.; Meyer, C.; Kohl, S. A. A.; Ballard, A. J.; Cowie, A.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Adler, J.; Back, T.; Petersen, S.; Reiman, D.; Clancy, E.; Zielinski, M.; Steinegger, M.; Pacholska, M.; Berghammer, T.; Bodenstein, S.; Silver, D.; Vinyals, O.; Senior, A. W.; Kavukcuoglu, K.; Kohli, P.; Hassabis, D., Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**.
7. Tunyasuvunakool, K.; Adler, J.; Wu, Z.; Green, T.; Zielinski, M.; Zidek, A.; Bridgland, A.; Cowie, A.; Meyer, C.; Laydon, A.; Velankar, S.; Kleywegt, G. J.; Bateman, A.; Evans, R.; Pritzel, A.; Figurnov, M.; Ronneberger, O.; Bates, R.; Kohl, S. A. A.; Potapenko, A.; Ballard, A. J.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Clancy, E.; Reiman, D.; Petersen, S.; Senior, A. W.; Kavukcuoglu, K.; Birney, E.; Kohli, P.; Jumper, J.; Hassabis, D., Highly accurate protein structure prediction for the human proteome. *Nature* **2021**.
8. Bahar, I.; Atilgan, A. R.; Erman, B., Direct evaluation of thermal fluctuations in proteins using a single-parameter harmonic potential. *Fold Des* **1997**, *2* (3), 173-81.
9. Atilgan, A. R.; Durell, S. R.; Jernigan, R. L.; Demirel, M. C.; Keskin, O.; Bahar, I., Anisotropy of fluctuation dynamics of proteins with an elastic network model. *Biophys J* **2001**, *80* (1), 505-15.
10. Yang, L.; Song, G.; Jernigan, R. L., Protein elastic network models and the ranges of cooperativity. *Proc Natl Acad Sci U S A* **2009**, *106* (30), 12347-52.
11. Tama, F.; Gadea, F. X.; Marques, O.; Sanejouand, Y. H., Building-block approach for determining low-frequency normal modes of macromolecules. *Proteins* **2000**, *41* (1), 1-7.
12. Dehouck, Y.; Mikhailov, A. S., Effective harmonic potentials: insights into the internal cooperativity and sequence-specificity of protein dynamics. *PLoS Comput Biol* **2013**, *9* (8), e1003209.
13. Tirion, M. M., Large Amplitude Elastic Motions in Proteins from a Single-Parameter, Atomic Analysis. *Phys Rev Lett* **1996**, *77* (9), 1905-1908.
14. Tama, F.; Sanejouand, Y. H., Conformational change of proteins arising from normal mode calculations. *Protein Eng.* **2001**, *14* (1), 1-6.

15. Tekpinar, M.; Yildirim, A., Only a Subset of Normal Modes is Sufficient to Identify Linear Correlations in Proteins. *J Chem Inf Model* **2018**, *58* (9), 1947-1961.
16. Kormos, B. L.; Baranger, A. M.; Beveridge, D. L., A study of collective atomic fluctuations and cooperativity in the U1A-RNA complex based on molecular dynamics simulations. *J Struct Biol* **2007**, *157* (3), 500-13.
17. Lee, S. Y.; Cho, H. S.; Pelton, J. G.; Yan, D. L.; Henderson, R. K.; King, D. S.; Huang, L. S.; Kustu, S.; Berry, E. A.; Wemmer, D. E., Crystal structure of an activated response regulator bound to its target. *Nat. Struct. Biol.* **2001**, *8* (1), 52-56.
18. Melvin, R. L.; Xiao, J.; Godwin, R. C.; Berenhaut, K. S.; Salsbury, F. R., Jr., Visualizing correlated motion with HDBSCAN clustering. *Protein Sci* **2018**, *27* (1), 62-75.
19. Calligari, P.; Gerolin, M.; Abergel, D.; Polimeno, A., Decomposition of Proteins into Dynamic Units from Atomic Cross-Correlation Functions. *J Chem Theory Comput* **2017**, *13* (1), 309-319.
20. Shudler, M.; Niv, M. Y., BlockMaster: partitioning protein kinase structures using normal-mode analysis. *J Phys Chem A* **2009**, *113* (26), 7528-34.
21. Lange, O. F.; Grubmuller, H., Generalized correlation for biomolecular dynamics. *Proteins* **2006**, *62* (4), 1053-61.
22. 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 Rio, J. F.; Wiebe, M.; Peterson, P.; Gerard-Marchant, P.; Sheppard, K.; Reddy, T.; Weckesser, W.; Abbasi, H.; Gohlke, C.; Oliphant, T. E., Array programming with NumPy. *Nature* **2020**, *585* (7825), 357-362.
23. Hunter, J. D., Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* **2007**, *9* (3), 90-95.
24. Zhang, S.; Krieger, J. M.; Zhang, Y.; Kaya, C.; Kaynak, B.; Mikulska-Ruminska, K.; Doruker, P.; Li, H.; Bahar, I., ProDy 2.0: Increased Scale and Scope after 10 Years of Protein Dynamics Modelling with Python. *Bioinformatics* **2021**.
25. Bakan, A.; Meireles, L. M.; Bahar, I., ProDy: Protein Dynamics Inferred from Theory and Experiments. *Bioinformatics* **2011**, *27* (11), 1575-1577.
26. Hagberg, A. A.; Schult, D. A.; Swart, P. J., Exploring Network Structure, Dynamics, and Function using NetworkX. *Proceedings of the 7th Python in Science Conference* **2008**, 11-16.
27. Gowers, R. J.; Linke, M.; Barnoud, J.; Reddy, T. J. E.; Melo, M. N.; Seyler, S. L.; Domanski, J.; Dotson, D. L.; Buchoux, S.; Kenney, I. M.; Beckstein, O., MDAAnalysis: A Python Package for the Rapid Analysis of Molecular Dynamics Simulations. *Proceedings of the 15th Python in Science Conference* **2016**, 98-105.
28. Michaud-Agrawal, N.; Denning, E. J.; Woolf, T. B.; Beckstein, O., MDAAnalysis: a toolkit for the analysis of molecular dynamics simulations. *J Comput Chem* **2011**, *32* (10), 2319-27.
29. Ichiye, T.; Karplus, M., Collective motions in proteins: a covariance analysis of atomic fluctuations in molecular dynamics and normal mode simulations. *Proteins* **1991**, *11* (3), 205-17.
30. de Groot, B. L.; van Aalten, D. M.; Scheek, R. M.; Amadei, A.; Vriend, G.; Berendsen, H. J., Prediction of protein conformational freedom from distance constraints. *Proteins* **1997**, *29* (2), 240-51.
31. Humphrey, W.; Dalke, A.; Schulten, K., VMD: visual molecular dynamics. *J Mol Graph* **1996**, *14* (1), 33-8, 27-8.
32. Schrodinger, LLC, The PyMOL Molecular Graphics System, Version 1.3r1. 2010.

33. Baker, F. N.; Porollo, A., CoeViz: a web-based tool for coevolution analysis of protein residues. *BMC Bioinformatics* **2016**, *17*, 119.
34. Hopf, T. A.; Green, A. G.; Schubert, B.; Mersmann, S.; Scharfe, C. P. I.; Ingraham, J. B.; Toth-Petroczy, A.; Brock, K.; Riesselman, A. J.; Palmedo, P.; Kang, C.; Sheridan, R.; Draizen, E. J.; Dallago, C.; Sander, C.; Marks, D. S., The EVcouplings Python framework for coevolutionary sequence analysis. *Bioinformatics* **2019**, *35* (9), 1582-1584.
35. Sethi, A.; Eargle, J.; Black, A. A.; Luthey-Schulten, Z., Dynamical networks in tRNA:protein complexes. *Proc Natl Acad Sci U S A* **2009**, *106* (16), 6620-5.
36. Negre, C. F. A.; Morzan, U. N.; Hendrickson, H. P.; Pal, R.; Lisi, G. P.; Loria, J. P.; Rivalta, I.; Ho, J.; Batista, V. S., Eigenvector centrality for characterization of protein allosteric pathways. *Proc Natl Acad Sci U S A* **2018**, *115* (52), E12201-E12208.
37. Botello-Smith, W. M.; Luo, Y., Robust Determination of Protein Allosteric Signaling Pathways. *J Chem Theory Comput* **2019**, *15* (4), 2116-2126.
38. East, K. W.; Newton, J. C.; Morzan, U. N.; Narkhede, Y. B.; Acharya, A.; Skeens, E.; Jogl, G.; Batista, V. S.; Palermo, G.; Lisi, G. P., Allosteric Motions of the CRISPR-Cas9 HNH Nuclease Probed by NMR and Molecular Dynamics. *J. Am. Chem. Soc.* **2020**, *142* (3), 1348-1358.
39. Clarke, D.; Sethi, A.; Li, S.; Kumar, S.; Chang, R. W. F.; Chen, J.; Gerstein, M., Identifying Allosteric Hotspots with Dynamics: Application to Inter- and Intra-species Conservation. *Structure* **2016**, *24* (5), 826-837.
40. Dokholyan, N. V., Controlling Allosteric Networks in Proteins. *Chem Rev* **2016**, *116* (11), 6463-6487.
41. Wodak, S. J.; Paci, E.; Dokholyan, N. V.; Berezovsky, I. N.; Horovitz, A.; Li, J.; Hilser, V. J.; Behar, I.; Karanicolas, J.; Stock, G.; Hamm, P.; Stote, R. H.; Eberhardt, J.; Chebaro, Y.; Dejaegere, A.; Cecchini, M.; Changeux, J. P.; Bolhuis, P. G.; Vreede, J.; Faccioli, P.; Orioli, S.; Ravasio, R.; Yen, L.; Brito, C.; Wyart, M.; Gkeka, P.; Rivalta, I.; Palermo, G.; Mccammon, J. A.; Panecka-Hofman, J.; Wade, R. C.; Di Pizio, A.; Niv, M. Y.; Nussinov, R.; Tsai, C. J.; Jang, H.; Padhorny, D.; Kozakov, D.; Mcleish, T., Allostery in Its Many Disguises: From Theory to Applications. *Structure* **2019**, *27* (4), 566-578.
42. Di Paola, L.; De Ruvo, M.; Paci, P.; Santoni, D.; Giuliani, A., Protein Contact Networks: An Emerging Paradigm in Chemistry. *Chem Rev* **2013**, *113* (3), 1598-1613.
43. Brandes, U., Fleischer, D., Centrality measures based on current flow. In *STACS'05: Proceedings of the 22nd annual conference on Theoretical Aspects of Computer Science*, Diekert, V. D., B., Ed. Springer: Germany, 2005; pp 533-544.
44. Zaki, M. J.; Meira, W., Data mining and analysis : fundamental concepts and algorithms. **2013**.
45. Girvan, M.; Newman, M. E., Community structure in social and biological networks. *Proc Natl Acad Sci U S A* **2002**, *99* (12), 7821-6.
46. Sheik Amamuddy, O.; Verkhivker, G. M.; Tastan Bishop, O., Impact of Early Pandemic Stage Mutations on Molecular Dynamics of SARS-CoV-2 M(pro). *J Chem Inf Model* **2020**.
47. Penkler, D. L.; Atilgan, C.; Tastan Bishop, O., Allosteric Modulation of Human Hsp90alpha Conformational Dynamics. *J Chem Inf Model* **2018**, *58* (2), 383-404.
48. Tekpinar, M.; Yildirim, A., Impact of dimerization and N3 binding on molecular dynamics of SARS-CoV and SARS-CoV-2 main proteases. *J Biomol Struct Dyn* **2021**, 1-12.
49. Yen, J. Y., Finding the K Shortest Loopless Paths in a Network. *Management Science* **1971**, *17* (11), 712-716.

50. Glykos, N. M., Software news and updates. Carma: a molecular dynamics analysis program. *J Comput Chem* **2006**, *27* (14), 1765-8.
51. Grant, B. J.; Rodrigues, A. P.; ElSawy, K. M.; McCammon, J. A.; Caves, L. S., Bio3d: an R package for the comparative analysis of protein structures. *Bioinformatics* **2006**, *22* (21), 2695-6.
52. Van Wart, A. T.; Durrant, J.; Votapka, L.; Amaro, R. E., Weighted Implementation of Suboptimal Paths (WISP): An Optimized Algorithm and Tool for Dynamical Network Analysis. *J Chem Theory Comput* **2014**, *10* (2), 511-517.
53. Pasi, M.; Tiberti, M.; Arrigoni, A.; Papaleo, E., xPyder: a PyMOL plugin to analyze coupled residues and their networks in protein structures. *J Chem Inf Model* **2012**, *52* (7), 1865-74.
54. Sheik Amamuddy, O.; Glenister, M.; Tshabalala, T.; Tastan Bishop, O., MDM-TASK-web: MD-TASK and MODE-TASK web server for analyzing protein dynamics. *Comput Struct Biotechnol J* **2021**, *19*, 5059-5071.