Journal of
CHEMICAL INFORMATION
AND MODELING

Article

pubs.acs.org/jcim

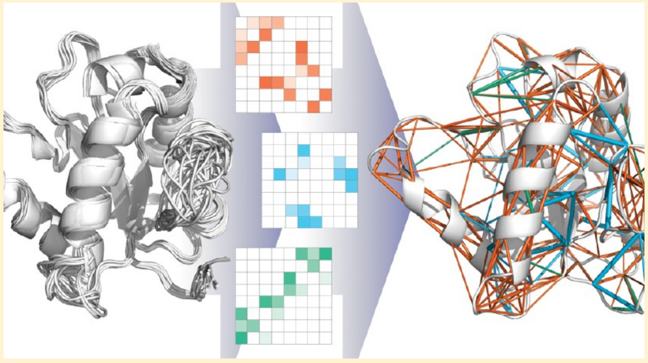
PyInteraph: A Framework for the Analysis of Interaction Networks in Structural Ensembles of Proteins

Matteo Tiberti,*,[†] Gaetano Invernizzi,[‡] Matteo Lambrughi,[†] Yuval Inbar,[§] Gideon Schreiber,[§] and Elena Papaleo*,[‡]

[†]Department of Biotechnology and Biosciences, University of Milano-Bicocca, Piazza della Scienza 2, 20126 Milan, Italy

[‡]Structural Biology and NMR Laboratory, Department of Biology, University of Copenhagen, Ole Maaløes Vej 5, 2200, Copenhagen, Denmark

[§]Department of Biological Chemistry, Weizmann Institute of Science, 761000 Rehovot, Israel



ABSTRACT: In the last years, a growing interest has been gathering around the ability of Molecular Dynamics (MD) to provide insight into the paths of long-range structural communication in biomolecules. The knowledge of the mechanisms related to structural communication helps in the rationalization in atomistic details of the effects induced by mutations, ligand binding, and the intrinsic dynamics of proteins. We here present *PyInteraph*, a tool for the analysis of structural ensembles inspired by graph theory. *PyInteraph* is a software suite designed to analyze MD and structural ensembles with attention to binary interactions between residues, such as hydrogen bonds, salt bridges, and hydrophobic interactions. *PyInteraph* also allows the different classes of intra- and intermolecular interactions to be represented, combined or alone, in the form of interaction graphs, along with performing network analysis on the resulting interaction graphs. The program also integrates the network description with a knowledge-based force field to estimate the interaction energies between side chains in the protein. It can be used alone or together with the recently developed *xPyder PyMOL* plugin through an *xPyder*-compatible format. The software capabilities and associated protocols are here illustrated by biologically relevant cases of study. The program is available free of charge as Open Source software via the GPL v3 license at http://linux.btbs.unimib.it/pyinteraph/.

1. INTRODUCTION

The network paradigm has been extensively used to describe the structure, topology, and dynamics of proteins.[1] Intramolecular noncovalent interactions between a pair of residues in a protein are crucial in determining protein structure and dynamics, and they can be collectively represented in the form of a network, namely a Protein Structure Network (PSN).[2−4] Studies in the PSN field pointed out the role of hub residues and other central elements in the PSN that can be connected to protein function, allosteric regulation, signal transduction, protein stability, and the effects induced by post-translational modification or binding with other biological partners.

In a PSN, the nodes of the network are generally the side chains of the protein residues, even if individual atoms can also be used. Edges in the network can be defined with different strategies, as for example distances between the side chains, atomic contacts, van der Waals interactions, etc., ...[1,3,5] Alternatively, interaction energy can be estimated with different methods, and the network can be constructed based on the interaction energy between each pair of residues in the protein.[6−9]

Protein structure networks are "small-worlds".[10−13] This is a crucial feature suitable for the fast transmission of conformational changes at distal sites. Indeed, in the small-world of PSNs, the amino acids can communicate with each other by the shortest paths available. Moreover, PSN are characterized by a small number of hubs compared to other kinds of networks, and the hub residues have generally an important role for protein stability or function.[1,12,14] The "signal" in protein structure during dynamics can propagate using multiple paths that have often nodes in common.[15−17]

Web servers, programs, or plugins are available to study protein networks from protein structure files in the Protein Data Bank (PDB), as for example the *RING* server,[18] *RINalyzer*,[19] *GraProStr*,[20] and *SPACER*.[21] Nevertheless, it is known that protein structures are better described as an ensemble of different conformational states in a dynamic equilibrium that can be perturbed by the binding with biological partners, post-translational modifications, or mutations.[22,23] Indeed, both experimental and computational methods that provide a structural ensemble of conformations in atomistic details, as for example NMR[24−27] or Molecular Dynamics (MD),[28−30] can be integrated to methods inspired by graph theory to investigate long-range structural communication and allostery. For example, the potential of NMR-derived parameters, as chemical shifts, has been exploited to derive paths of allosteric communication in proteins.[25,26,31,32]

Received: October 30, 2013

Not only, also contact networks involving conformationally heterogeneous residues can be directly detected by X-ray crystallographic data.[33] Thus, the progress in the PSN field and their description in a dynamic framework are likely to enrich the knowledge on functional conformational changes in proteins.

To identify paths of long-range communication in structural ensembles of proteins, a number of methods is currently employed.[9,34−41] Recently, softwares to calculate PSNs or to describe structural communication in protein ensembles have been also made available. Some of them are mainly inspired by concepts from engineering, and they describe the protein as a mechanical construct rather than a chemical molecule. In this class, we can find the so-called force distribution analysis (FDA), which is based on the analysis of forces intercurring between atoms.[42] FDA is generally able to identify structural communication even in stiff structures in which communication is very subtle and occurs without evident atomic displacement. One limitation of the method is that it can be applied to those ensembles for which acting forces can be calculated, as for example MD-derived ensembles. Another very recent package is *GSATools*,[43] which describes the structures of the ensemble in terms of a structural alphabet. The alphabet describes 25 canonical states of four-residue protein fragments. Once each conformation is encoded as a string, the string collection can be further analyzed to extract information about paths of communication.

PSN-derived methods are implemented for example in the *Wordom* toolkit for MD analysis[44] or in the recently released PSN-ensemble.[9] Indeed, this class of PSN approaches is based on the description of the more persistent atomic contacts between side chain atoms during dynamics to define nodes and connections in the graph. The aforementioned PSN-based methods consider the atomic contacts and thus account for van der Waals effects, which are often playing a crucial role in long-range communication.[3] Nevertheless, they do not discriminate the different chemical-physical properties of the residues. Thus, graph analysis approaches applied to the network of intramolecular interactions (IIN), describing the class of most relevant weak interactions occurring in proteins, may finely complement the PSN information. Indeed, weak interactions in proteins, as for example hydrogen bonds, can be important components of the intra- and intermolecular communication. To have a comprehensive view on protein structural communication, IIN description can be integrated with the PSN approaches based on noncovalent atomic contacts.[9,43,44]

In this scenario, we here provide a versatile *Python* tool (*PyInteraph*) that is tightly connected to the *MDAnalysis* package[45] to describe IIN from structural/dynamic ensembles. The program also allows each class of weak interactions to be separately described or integrated in a macronetwork of interactions. Indeed, a remarkable feature of *PyInteraph* is that, similarly to the PSN methods mentioned above, it employs information on the atomic coordinates to build a network representation. On the other side, differing from the classical PSNs, it does this by calculating specific classes of interaction, which are the ones more commonly found in protein ensembles and important for protein architecture and dynamics.

Moreover, a method to provide a description of the interaction energy by pairs of residues was included in *PyInteraph*, implementing the *Hunter* statistical potential recently developed by Schreiber's group[46,47] and here used for the first time to describe interaction networks. This potential allows us to estimate the interaction energy between the side chains of each pair of protein residues, using four-distances defined between two sets of atoms each of them belonging to one of the residues in the pair. *PyInteraph* also allows performing networks analysis (i.e., calculation of hubs, connected components, paths of communication) on the interaction graphs.

PyInteraph outputs have been made compatible with our recently developed *PyMOL* (https://www.pymol.org) plugin, *xPyder*,[48] to visualize the results on a reference three-dimensional (3D) structure. Nevertheless, since *xPyder* was conceived to plot only a pairwise relationship between Cα atoms, we also included in *PyInteraph* a *PyMOL* plugin, *interaction_plotter* to plot the results on a reference 3D structure considering each atom involved in the interaction. This is especially important for hydrogen bonds. We here illustrate *PyInteraph*, along with some applications (as examples) to MD structural ensembles in the attempt to complement and rationalize available experimental data on different target proteins.[49−53]

2. METHODS

2.1. Class of Interactions That Are Employed in *PyInteraph*. Three classes of interactions are included in the program: hydrophobic interactions, salt bridges, and hydrogen bonds (H-bonds). For each class geometric criteria are defined to evaluate if the interaction is present between selected pairs of atoms or atom groups in a given protein conformation. The persistence for each pairwise interaction is then calculated as the fraction of the number of structures of the ensemble in which the interaction was observed.

The criteria to define a pair of interacting residues for each structure can be modified by the user for each class of interactions. Nevertheless, we encourage the user to refer to the literature in terms of relevant cutoffs to define an interaction and to use criteria that can be justified and find a fundament in the structural biology field.

For hydrophobic contacts, the interaction between two residues is included if the center of mass of the side chain of the two hydrophobic residues is found within 5 Å of distance as a default. Default residues to be considered for hydrophobic interactions are Ala, Ile, Val, Leu, Phe, Met, Trp, and Pro. The list can be modified by the user to include a specific subset of residues for the analysis.

Since centers of mass are considered for the analysis, the mass of each single atom has to be taken into account. Each MD force field is known to have different mass definitions, thus the user has the opportunity to specify one of the several mass databases that come prepackaged with *PyInteraph*, belonging to the GROMOS, AMBER, CHARMM, ENCAD, and OPLS families. This is especially important when considering united-atom force fields, such as GROMOS. Finally, if nonstandard atoms are used and their masses are not specified in the mass repository, their atomic masses are guessed on the base of the atom names and the program provides a warning message.

For salt bridges, all the distances between atom pairs belonging to two "charged groups" of two different residues are calculated, and the charged groups are considered as interacting if at least one pair of atoms is found at a distance shorter than 4.5 Å as a default. In Asp and Glu, the atoms that form the carboxylic group are considered (including both the carbon and the oxygen atoms). For Lys, Arg, and His (only if protonated) the NH_3^+, the guanidinium group, and the imidazole ring are employed, respectively. In the case of His, *PyInteraph* determines if the hydrogen is presented at both the Nε and Nδ of the His residue (i.e., HISH atom type in several force fields) from the input topology. Otherwise, all histidine residues are considered as neutral. Moreover, in the default charged groups the N- and C-terminal of the protein are positively and negatively charged, respectively. Those default charged groups are customizable by the user.

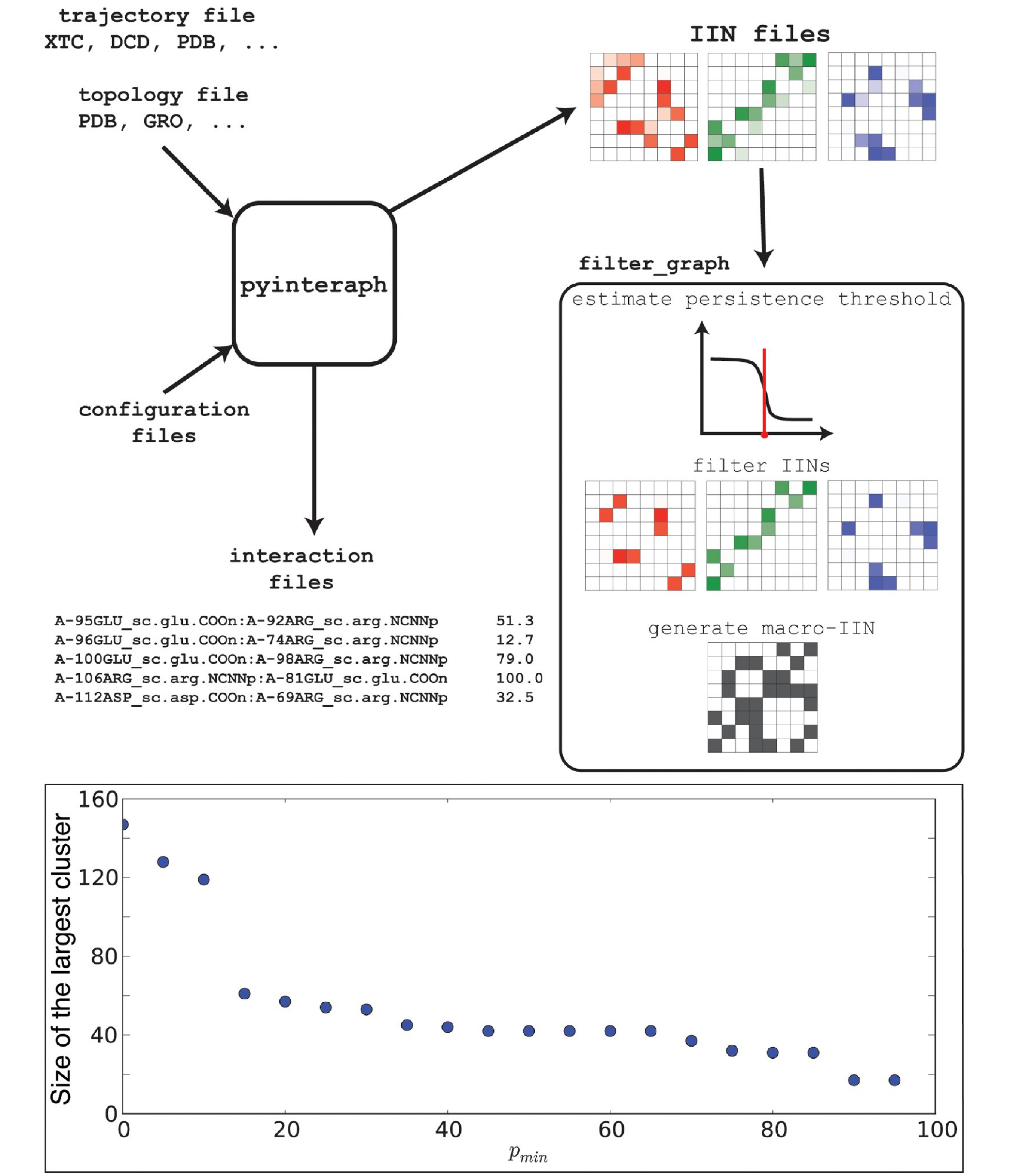
The module for salt-bridge interactions also allows the calculation of repulsive interactions (i.e., positively charged vs positively charged or negatively charged vs negatively charged residues). In the *.ini* configuration file, the positively charged groups have to be defined as ‘p’ and the negatively charged as ‘n’.

A H-bond is identified when both the distance between the acceptor atom and the hydrogen atom is lower than 3.5 Å and the donor-hydrogen-acceptor atom angle is greater than 120°. These default parameters can be modified by the user. As a default, both side chain and main chain H-bonds are included. The groups can be modified by the user allowing to select only side chain-side chain, main chain-main chain, or main chain-side chain H-bonds, and the acceptor and donor atoms have to be specified in the configuration file.

2.2. Calculation of the Intra- or Intermolecular Interaction Networks (IIN) for Individual Classes of Interaction. At first, the program identifies all the pairwise interactions for the selected class (i.e., hydrophobic interactions, H-bonds, salt bridges), and it associates with each pair a value of persistence of the interaction in the ensemble. Afterward, these interactions are merged together in an Intra- or Intermolecular Interaction Network (IIN) that aims at providing an integrated view of the interaction class in the structural ensemble. Different IINs can be defined for each interaction class, as well as they can be then combined in a more complex graph, as explained in Section 2.4. In details, the IIN is defined as a graph, in which residues are the nodes of the graph and each edge represents a specific interaction between them. Notably, the network is defined as having one edge per residue pair. Indeed, only one edge is considered in the case of pair of residues with multiple atoms involved in the interaction. Starting from the definition of the different interaction classes explained above (Section 2.1), it is necessary to define one persistence value per residue pair (i.e., the edge weight for that interaction). This is especially relevant for H-bonds since one or more of these interactions may simultaneously exist between two residues. Considering how the charged groups for salt bridges are conceived, the same issue can occur if positively or negatively charged residues are located at the N- and C-terminal extremities, respectively, and both the side chain and the terminal group are interacting with another oppositely charged amino acid in the protein. In these cases, *PyInteraph* calculates, for each conformation in the structural ensemble, an edge between two given residues if at least one atomic interaction of the selected class is identified. The persistence value is then calculated as the ratio between the number of structures in which an edge was identified (i.e., in which at least one interaction was present between the two residues) and the total number of structures in the ensemble. The edge weight of the IIN between two given residues is thus calculated so that it may or may not coincide with the edge value for the individual interatomic interaction, depending on the number of conformations of the ensembles in which the interatomic interactions between the two residues are simultaneously or exclusively present. For instance, if a pair of residues can form up to two H-bonds involving different atoms of each residue and these H-bonds are correlated in the structural ensemble, they will always be identified in each conformation and the edge value of the IIN will coincide with the value of their individual persistence. On the other hand, if the two aforementioned H-bonds are completely uncorrelated, they will never be present in the same structure, and the persistence of the IIN edge will be equal to the sum of their persistency values. The hydrophobic-interaction networks do not present this kind of issues since we defined them on the base of pairwise interactions. Indeed, in hydrophobic interaction networks, the distance between the centers of mass of the residue side chains is employed, and there are no issues related to contacts between multiple atoms of the residues involved.

2.3. Calculation of the Persistence Cutoff for Significant Interactions. *PyInteraph* procedure requires the user to filter the interactions in the individual IIN to exclude very transient interactions, which cannot account for important structural or functional features and are likely to be related, for example, to the sampling of rare conformations during dynamics under the force-field description. We thus implemented the so-called p_{crit} calculation, which is derived from known properties of PSNs[3,54−56] and is based on the size of the largest interaction connected component (i.e., cluster) identified at different persistence values. In graph theory a connected component is defined as a subgraph in which a path exists between any two vertices, but no paths exist to any other vertices of the main graph, meaning that there are no edges connecting two connected components. The size of the largest cluster in the graph is generally employed to understand the nature and properties of graphs.[3,54−56] In the case of protein structures, it has been also observed that a critical value exists below which the residues in the PSN are almost completely connected (resulting thus in just one large cluster), and above this critical value the PSN splits into smaller clusters.[56] The transition is very sharp and occurs on a narrow range of cutoffs, and it is generally used to define the threshold of significance (I_{crit}) for the edges in the graph. Analyses of PDB structures showed that the I_{crit} falls in a narrow range for proteins of different sizes and folds, suggesting that it can be used as a general method for threshold detection.

Since *PyInteraph* uses atomic contacts selected on the base of noncovalent interactions, it provides a graph with weighted edges, which is conceptually very similar to the resulting graph of a classical PSN. The *filter_graph* tool of *PyInteraph* thus calculates a collection of graphs from the interaction map of the same data set (i.e., the same structural ensemble) provided in input (i.e., the individual IIN) for different persistence thresholds (called p_{min} for consistency with the I_{min} by Brinda et al.[56]) (Figure 1, lower panel) and calculates for each of them the size of the largest connected component (i.e., largest cluster). Each graph is then filtered by removing all the edges that have a weight lower than the chosen p_{min}. The analysis is iteratively carried out for increasing p_{min} values. As a result, each graph derived at a specific p_{min} value will feature a different number of edges. For each of these graphs the clusters are calculated and the size of the largest cluster will thus decrease (or at most remain equal) with increasing p_{min} values (i.e., increasing thresholds) since more and more edges will be filtered out. The significance threshold for interaction persistence can be then calculated as the value at which a sharp transition in the size of the largest cluster occurs (p_{crit}). Choosing p_{crit} as the threshold value allows removing many low-value edges that would increase noise and connect all the clusters into a single one, as would happen if no threshold is used. Conversely, keeping only very high-value edges would affect the overall network structure, leaving only highly interconnected clusters. Choosing the p_{crit} value for analyses allows us at the same time to filter out meaningless interactions and to maintain the network structure.



```
A-95GLU_sc.glu.COOn:A-92ARG_sc.arg.NCNNp      51.3
A-96GLU_sc.glu.COOn:A-74ARG_sc.arg.NCNNp      12.7
A-100GLU_sc.glu.COOn:A-98ARG_sc.arg.NCNNp     79.0
A-106ARG_sc.arg.NCNNp:A-81GLU_sc.glu.COOn     100.0
A-112ASP_sc.asp.COOn:A-69ARG_sc.arg.NCNNp     32.5
```

Figure 1. General workflow of the *PyInteraph* main tools. The lower panel also shows an example of the evolution of the size of the largest cluster in the graph as a function of the p_{min} value.

2.4. Calculation of the Macro-IIN. Since each of the individual IIN is constructed in a similar way (i.e., they have a similarly shaped adjacency matrix), they can be combined to form a comprehensive graph (macro-IIN) retaining contributions from the different single graphs. To select the interactions that have to be integrated in the macro-graph, a cutoff of interaction persistence has to be selected by *filter_graph*, as explained in Section 2.3 for each individual IIN.

In the selection of pairwise connections to include in the IIN matrix, two residues are considered as connected by H-bonds if at least one H-bond is present between two atoms of those residues. The full macro-IIN is then computed building an unweighted graph in which an edge between two residues (nodes) is present if an interaction of at least one of the three classes existed between them. Since the macro-IIN is unweighted, it has to be used only to provide an overall view of the more persistent interactions in the MD ensemble; it cannot be used itself for network analysis. *PyInteraph* then also includes a procedure to weight the macro-IIN according to the energy map values (defined in Section 2.5), which is the only macro-IIN that can be used for graph-analysis purposes.

2.5. Interaction Energy Maps. *PyInteraph* integrates a module to estimate the interaction energy between a pair of side chains, employing a component of a knowledge-based potential, called *Hunter*,[46] which is based on a four-distances description of the side-chain interactions.[47] Briefly, the potential has been defined on sets of four interatomic distances. These distances have been defined independently for each possible pair of residues (except glycine) by selecting two representative atoms per residue pair and calculating the four inter-residue distances between them. This allowed the possibility that the same amino acid is in contact with different partners via different atom pairs. The representative pairs were chosen as those having the largest number of contacts in a data set of 9394 high-resolution protein structures from the Protein Data Bank (PDB).[46] Once the four-distances set has been defined for four atom pairs, four-dimensional (4D) histograms with a constant bin size of 0.5 Å along each dimension from 0 to 10 Å were built, by measuring the relevant distances in the available structural data set to estimate the approximate probability distributions, followed by a smoothing step. The "pseudo-energy" values for each interaction are then calculated using the reversed Boltzmann equation

$$\Delta E = -k_b T \log(P_{real}/P_{rand})$$

where

$$P_x = P(\{dist\}|AA)^* P(AA)$$

$P_x(\{dist\}|AA)$ is the probability of observing the four-distance combination for a given residue pair, and $P(AA)$ is the probability to observe a side-chain contact for a given pair of residues in protein structures. P_{real} refers to the distribution encoded in the 4D histogram for the target pair of residues, while P_{rand} is built upon a random model which represents a scenario in which the side-chain conformations are not dictated by forces characteristic of a real protein. k_bT is defined as unity by default according to ref 47, but its value can be modified by means of a simple command-line option. The user can change the value to explicitly take into account the temperature at which the structural ensemble was derived and a specific value for the Boltzmann constant (k_b) to express the pseudoenergy in the desired unit.

The interaction pseudoenergy is calculated by *PyInteraph* for all the structures of the ensemble and then averaged for each residue pair. In this way, a single value per residue pair is obtained, resulting in an interaction matrix that can be used as a graph adjacency matrix, which features a single value per residue pair. In principle, the p_{crit} analysis carried out by the *filter_graph* module was designed to deal with the interaction persistence in the aforementioned IIN graphs but can be also applied to the network derived by interaction energies. Moreover, the macro-IIN described in Section 2.4 can be combined to the energy map, so that the pseudoenergy values for each pair of residues can be used as weights for the edges between the nodes of the macro-IIN.

2.6. Graph Analysis. The *PyInteraph* package includes also a network analysis module, named *graph_analysis* to postprocess the IINs or the weighted macro-IIN. The program is then also able to work on any graph provided in the adjacency matrix file format compatible with *PyInteraph*. The *graph_analysis* tool performs three different types of analysis on the selected graph. First, it identifies highly connected residues on the network, also called hubs. It does so by calculating the number of connections for each node and considering as hubs only those nodes having a number of connections higher than or equal to a user-selected threshold *k*. In PSN applications, it has been shown that hubs are generally residues connected by more than 3 or 4 edges,[1,12,14] so we recommend to employ these values as thresholds. *graph_analysis* not only defines the hubs but also provides for each of them the connectivity degree, which can be useful information to compare networks derived for example from different protein variants. Second, *graph_analysis* identifies connected components, which are isolated regions of the graph, as detailed in Section 2.3. Third, *graph_analysis* can calculate the shortest paths between two specified residues in the graph, using a variant of the depth-first search algorithm. While the default output is textual only, providing a reference PDB file to the script allows *graph_analysis* to write information about the identified connected components or hubs in the B-factor field of the input PDB file. Also, the identified paths can be independently saved as adjacency matrix files that can then be plotted by *xPyder PyMOL* plugin.[48]

2.7. MD Simulations of Target Proteins. Most of the simulations employed here as cases of study are taken from already published works, as described in the following. In particular, the 100 ns MD runs of wt and V16A *Aeropyrum pernix* Acylaminoacyl peptidase (ApAAP) are described in ref 40. The 200 ns concatenated trajectory of the cold-adapted alkaline phosphatase from *Vibrio proteinase* (VAP) is described in ref 57. The intrinsically disordered domain of Ataxin-3 ($AT3_{182-291}$) was investigated collecting overall 500 ns of MD in the recently published work by Invernizzi et al.[51] The case study of the interaction between a phosphovariant of Cdc34 and ubiquitin (Ub) in 40 ns MD simulations is described in ref 58.

Additional simulations were included in the present manuscript and not published elsewhere, and they were carried out by GROMACS v.4.6 (www.gromacs.org). In particular, for p53, we used a subset of MD simulations collected for a manuscript presently under preparation. The X-ray structure of the p53 DNA-binding domain (DBD, PDB entry 1TSR, chain B, residues 95−289)[59] in complex with DNA (PDB entry 1TSR, chains E and F) was used as starting structure for the simulations. The simulations were carried out using CHARMM27 (CHARMM22 with the CMAP) for 100 ns in the NPT ensemble at 300 K and 1 bar. Electrostatic and van der Waals interactions were truncated at 9 Å.

The E2 enzymes Ubc1 (PDB entry 1FXT, chain A) was simulated for 50 ns with the CHARMM22* force field in the NPT ensemble at 300 K and 1 bar. Electrostatic and van der Waals interactions were truncated at 10 Å.

All the p53 and E2 simulations were carried out using periodic boundary conditions and a dodecahedral box and TIP3P water models (www.gromacs.org). The LINCS algorithm[60] was used to constrain the bond lengths of heavy atoms, allowing the use of a 2 fs time-step. Long-range electrostatic interactions were calculated in all the cases using the Particle-Mesh Ewald (PME) summation scheme.[61]

3. RESULTS AND DISCUSSION

3.1. PyInteraph. *3.1.1. PyInteraph Overview.* The *PyInteraph* package allows the calculation of intra- or intermolecular interactions, such as salt bridges, hydrogen bonds, and hydrophobic interactions in structural ensembles (Figure 1). A knowledge-based potential is also available to estimate the interaction energy between residue side chains. The calculated interactions can be merged in networks of interactions of a specific class (sb-IIN, hb-IIN or hc-IIN, for salt bridges, H-bonds, and hydrophobic interactions, respectively) or in a macro-IIN (see Sections 2.2 and 2.4, respectively). The outputs can be analyzed with methods inspired by graph theory by the tool *graph_analysis*. In particular, the *graph_analysis* module calculate several features of the calculated IINs, such as hubs, connected components and paths between pairs of residues (see Section 2.6 for details).

The package includes user-editable configuration files and support files, as those containing the per-atom mass information on many popular MD force fields, the definition of charged groups, of acceptor and donor atoms for H-bonds, and the definition of the knowledge-based potential *Hunter*.[46,47]

The software suite is written in *Python* and *C*. It is mainly composed of a *Python* library, which can be integrated in external code, and few front-end scripts that perform the main calculations and further analysis. The most time-consuming parts of the calculations are handled by a highly efficient *C* library, which is wrapped for the *Python* interpreter using the *Cython* programming language.

The *PyInteraph* main script is a command-line tool that performs most of the analyses on the protein ensembles, outputting different files which contain data both for the single interactions and the interaction graphs, as detailed in Section 2.3. The *filter_graph* script can be used to postprocess the *graph files* obtained by the main program. This module allows the user to deal with the estimation of a significance threshold for the persistence of the interactions and to filter the graph of interest (both the individual or the macro-IIN) according to the selected threshold, as well as to prepare the macro-IIN merging the individual IIN graphs.

Finally, the *PyInteraph* package includes a basic *PyMOL* plugin, called *interaction_plotter*, which has been designed to map the identified interactions on the 3D structure of the target proteins.

3.1.2. System Requirement and Installation. *PyInteraph* requires a working installation of *Python 2.7* and few freely available scientific open source libraries, which are easily installed on the most common operating systems, such as OSX and Linux distributions. The required libraries include *MDAnalysis* 0.7.7, *Numpy 1.6*, *Networkx 1.5*, *Scipy 0.10.1*, and *Matplotlib 1.1*. The *interaction_plotter PyMOL* plugin requires a complete *PyMOL* installation, version 1.3 or above. We are aware of the fast evolution of *Python*, thus we are willing to support and update our software so that it will remain compatible with as many future and present *Python* versions as possible. Indeed, a transition to *Python* 3.3 is planned as soon as full compatibility of the libraries required for *PyInteraph* will be available.

The installation of *PyInteraph* is performed through a *distutils* setup script, which can install the novel *Python* module, its *C* extension and the front-end scripts. The location of the *PyInteraph* directory can be stored in a specific system variable that the program uses to automatically identify the localization of configuration and support files. Finally, the *interaction_plotter* plugin must be installed in *PyMOL* by means of the plugin handling interface.

3.1.3. Input Format. *PyInteraph* is able to analyze structural ensembles from several different sources thank to its tight connection with *MDAnalysis*,[45] as described above. It supports the most common MD trajectory formats, such as GROMACS, CHARMM, NAMD, LAMMPS, and AMBER file formats. It supports the most common plain text topology and coordinate formats from the aforementioned programs, such as .gro, .crd, .xyz, and .pdb files. Single and multiple-model PDB files can also be provided to the script as trajectory files to perform the graph analysis on individual structures, NMR-derived structural ensembles or any other ensembles of structures derived by computational or experimental methods.

For the proper functionality of the program, the trajectories should be formatted so that the molecules under analysis are contiguous (i.e., the periodic boundary conditions, if present, should be removed). Moreover, the program should be used on protein ensembles reflecting the sampling of a relatively local conformational basin, so that meaningful averages can be extracted. This is especially important in the case of the graph derived applying the knowledge-based potential, due to its high sensitivity to the local structural features of the protein conformations.[46,47]

3.1.4. Output Format and in-Depth Analyses. *PyInteraph* provides two main outputs for the individual IIN. (i) An *interaction file* includes a list of each pairwise interaction and the associated persistence in the structural ensemble. It contains the chain ID, residue number, residue ID, and the corresponding atoms or groups for each of the residue in a pair. The *interaction file* can be used to map on the 3D protein structure the interactions and their network by a *PyMOL* plugin (*interaction_plotter*), which is included in the *PyInteraph* software package. Interactions are plotted by connecting the relevant atoms or groups by sticks of thickness and shade of colors proportional to the interaction persistence. (ii) The second format, namely *graph file*, is a simple ASCII square matrix of persistence values separated by spaces. It represents the adjacency matrix for the macro- or the individual IIN graphs. It has been designed to be fully compatible with the *xPyder* plugin[48] that provides an interface to visualize any set of structural data that can be represented in a matricial format.

3.1.5. Customizability and Configuration Files. *PyInteraph* has been designed to support the most common residue types and classes of interactions. The software also supports new force-field masses, charged groups, noncanonical residues (e.g., residues modified by post-translational or chemical modifications), and nonprotein molecules thank to an user-editable configuration file (*.ini* file).

The user can also modify several parameters of the program by command-line arguments, such as distance, angle, and persistence cutoff values. For hydrophobic interactions, the user can specify the list of residue to include. An alternative application of this tool is the use of all the 20 amino acids to calculate an interresidue contact map. The information about force-field masses is stored in files, which are written in the standard JSON format, so that support for new molecules and force fields can be easily added. It is also possible to generate the mass-files from the standard GROMACS force-field files using a provided script (*parse_masses*).

In summary, the possibility to customize the configuration files and the program options improve the flexibility of the tool, allowing the user to include in the analysis nonstandard residues, generic customized nonprotein ligands, and force-field masses of different sources.

3.2. Applications. Some practical examples are discussed in the following to better illustrate the capabilities of *PyInteraph.* The applications here discussed are not exhaustive of the use of the package and its tools but are intended to provide some cases of study and inspiration for the user in terms of how the results can complement or integrate experimental data. Indeed, a "real-life" application can be the identification of persistent interactions in an ensemble framework for a target residue or a group of residues for experimental research, as well as to estimate the associated interaction energy. This can be useful, for example, to design experimental mutagenesis. It can also provide a structural rational on the local and long-range effects observed experimentally upon a mutation or protein modification, for example comparing with *PyInteraph* structural ensembles (experimental or simulated) of both wild-type and mutant/modified variants.

3.2.1. Application-1: Identify Relevant Pairs of Interactions in Structural Ensembles. *PyInteraph* has been mainly designed to provide an overall description in terms of networks of intra- and intermolecular interactions in a structural ensemble (in particular MD ensembles but not only). Nevertheless, a first and obvious application of the tools is the evaluation of a specific pairwise interaction or a well-defined group of local interactions. Indeed, once the *interaction file* with the persistence is calculated, the estimation of the target interactions can be derived directly from this file, where all the pairs of atoms/residues involved in the selected interaction class are listed along with their persistences. The format of these outputs is straightforward to both read and parse. This output can also be used by the *interaction_plotter* plugin to visualize the networks. Moreover, the *graph file* can be directly used as input file for *xPyder*[48] exploiting the filtering option of this plugin. The target interactions can thus be mapped on the 3D protein structure. To better illustrate this application, we here reported some cases of biologically relevant pairwise or localized intramolecular interactions, illustrating examples for H-bonds and hydrophobic interactions.

The hydrophobic interactions generally play a crucial role in the stabilization of the protein core and in the maintenance of the 3D structure and stability.[62] Hydrophobic and aromatic residues are usually highly packed inside the protein and shielded from the solvent, as in the p53 DNA Binding Domain (DBD).[59] Nevertheless, the p53-DBD has a β-sandwich fold, but it is naturally unstable and melts slightly above the body temperature, becoming prone to be inactivated by oncogenic mutations.[63] It has been proposed that p53-DBD has evolved to be naturally unstable, and this is essential for its activity and regulation.[50] NMR experiments showed several buried polar groups in p53-DBD, especially tyrosine residues that can be flexible and involved in the formation of suboptimal H-bond networks, determining its instability.[50] We here employed *PyInteraph* to investigate the role of those tyrosine residues in the stabilization of the p53-DBD, calculating the hydrophobic and H-bond interactions and their local networks from an MD ensemble and postprocessing the data with *xPyder*.[48] The p_{crit} functionality of *filter_graph* was used to define a threshold of significance for the interaction persistence (20%). Our analyses shows a stable network of hydrophobic interactions (Figure 2-A1) in agreement with experimental data that suggest its critical role in the formation of the protein core.[50] Moreover some Tyr residues (Tyr-163, Tyr-205, Tyr-236) turned out to be localized in key positions in the structure and probably involved in interactions between the β-sheets, L2 and L3 loops (Figure 2-A2), in agreement with the experiments.[50] Indeed, the hb-IIN calculated with *PyInteraph* and exploiting the *interaction_plotter* plugin shows that a highly persistent H-bond (more than 90%) is present between Thr-253 Hγ2 and Tyr-236 Oμ in our MD ensemble in agreement with the main conformers from the NMR ensemble (Figure 2A-3).[50]

Another example of functionally relevant H-bonds here provided is related to the E2 ubiquitin (Ub)-conjugating enzymes. E2 enzymes have a central role in the Ub-mediated proteolysis of proteins as they transfer Ub or Ub-like proteins to the target substrate.[64] Despite being extensively investigated, their catalytic mechanism remains elusive, and it is mainly related to a conserved His-Pro-Asn motif (HPN) upstream of the catalytic cysteine, which also exhibited different arrangements in the solved X-ray structures of E2s. A recent work, in which chemical shifts data, temperature coefficient measurement derived for the human E2 HIP2, along with analysis of all the known structures of E2 were integrated, identified a conserved H-bond in the HPN motif. The H-bond involves the imidazole ring of His and the amide proton of the backbone of HPN Asn. It is a critical element for the orientation of the His of HPN.[49] Herein, we calculated by *PyInteraph* the H-bond network on a 50 ns MD simulation of *Saccharomyces cerevisiae* Ubc1, an E2 enzyme homologous to HIP2. In particular, we employed the *interaction_plotter* plugin to visualize the H-bond networks at atomic level (Figure 2-B1). Our results show the presence of a highly persistent (100%) H-bond between the amide proton of the Asn and the imidazole ring of the His in the HPN motif (Figure 2-B2) in agreement with the recent experimental findings, further supporting the relevance of this interaction for the proper orientation of the His ring.[49]

3.2.2. Application-2: Networks of Salt Bridges or Hydrogen Bonds. Electrostatic interactions, and salt bridges in particular, may play a crucial role in protein stability, and they can exert both local and long-range effects.[65,66] Salt bridges are indeed highly flexible and cooperatively organized in networks across the protein structure. Differences in salt-bridge networks have been often associated with enzymes from differently temperature-adapted organisms.[67] In this context, the intramolecular interactions are only one side of the story. Indeed, proteins involved in many fundamental processes as well as many extremophilic enzymes so far identified are organized in intermolecular complexes, i.e. multimeric.

Here we present an example of a salt-bridge based IIN in the dimeric structure of the cold-adapted phosphatase from *Vibrio proteinase* (VAP).[57] In particular, we calculated all the salt bridges both intra- and intersubunits, and we built a network from those interactions by *PyInteraph*. The p_{crit} calculation was also employed to define a significant threshold of interaction persistence of 20%. The sb-IIN was then analyzed with *xPyder*[48] filtering tools, focusing the attention on the ion pairs at the interface between the two monomers (Figure 3A). It turned out, in agreement with the previously published results,[57] that the enzyme is characterized by a low number of dispersed electrostatic interactions at the interface, a typical feature of cold-adapted enzymes.[67]

Salt bridges and electrostatic interactions may also play a role for structural and functional properties of proteins belonging to the class of Intrinsically Disordered Proteins (IDPs).[68] In this context, we recently investigated the conformational ensemble of the disordered region of ataxin-3 (AT3$_{182-291}$) by biophysical spectroscopies and MD simulations.[51] It turned out that the domain can populate two different conformational states with a

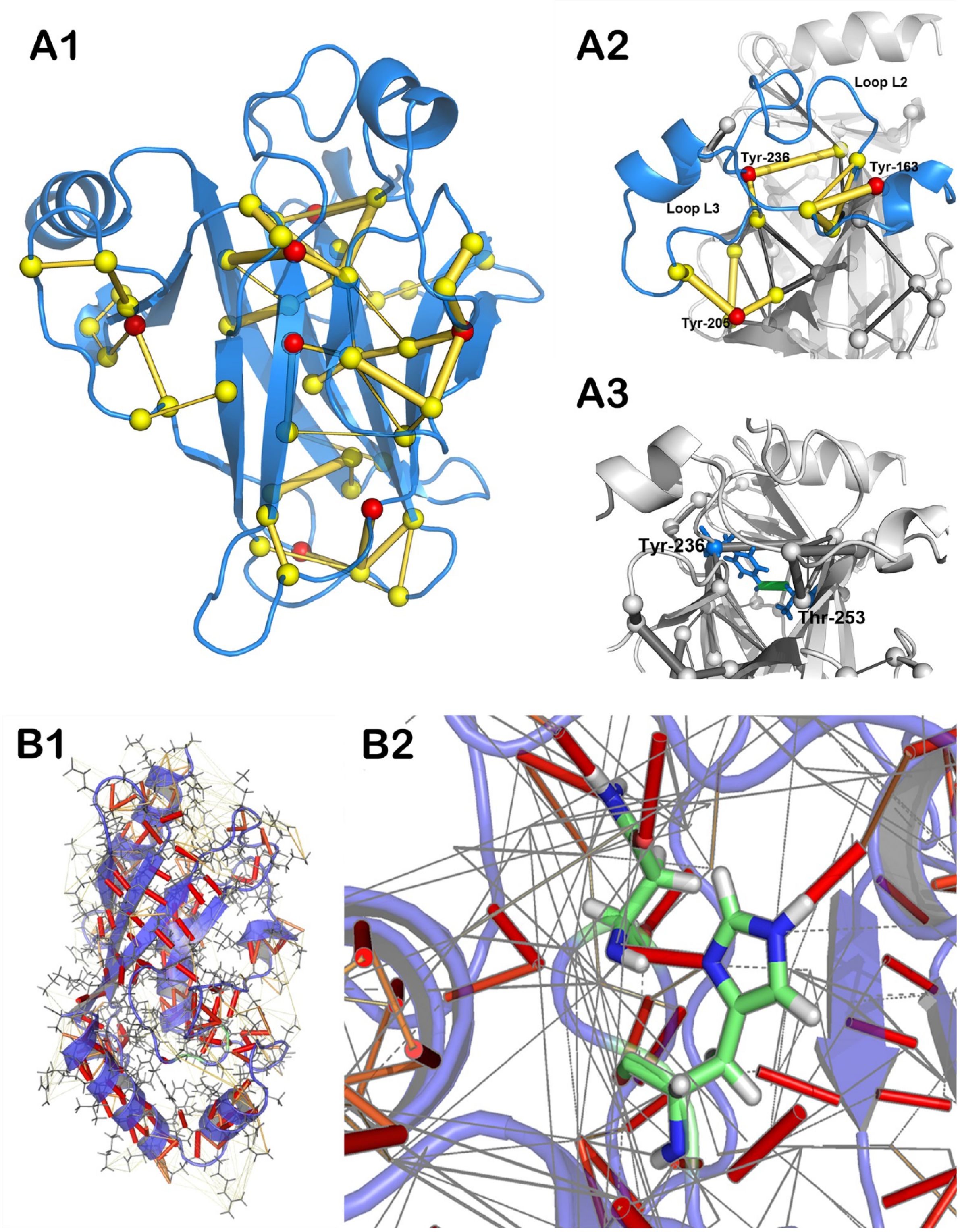


Figure 2. A) Hydrophobic interactions involved in the core of the p53 DNA binding domain (DBD). The hydrophobic interactions of the MD ensemble of p53 DBD were calculated with *PyInteraph* and then postprocessed with *xPyder* filtering interactions with persistence lower than 20%. The p53 DBD structure is shown in blue cartoon. The Cα atoms of the residues involved in the interactions and of the tyrosines are indicated as yellow and red spheres, respectively (1,2). The hydrophobic interactions are represented as cylinders connecting the Cα atoms of residues and their thickness is proportional to the persistence value (A1). Zoom on the interactions that are suggested to stabilize the loop L2 and loop L3 conformations (A2). H-bond between Thr-253 and Tyr-236 (A3). The two residues are highlighted in blue and their side chains are represented as sticks. The H-bond is represented as a green cylinder connecting the Thr-253 Hγ2 and Tyr-236 Oμ atoms using the *interaction_plotter PyMOL* plugin provided within *PyInteraph*. B) Conserved H-bond in the HPN motif of E2 enzymes. H-bonds of a 50 ns MD simulation of the E2 enzyme Ubc1 were calculated by *PyInteraph* and plotted with its integrated *PyMOL* plugin, *interaction_plotter* (B1). Cylinder thickness is proportional to the persistence value of the atoms involved in the H-bonds. A close view on the His and Asn residues in the HPN motif (B2) revealed a H-bond with high persistence (100%) between the amide proton of the Asn and the imidazole ring of the His.

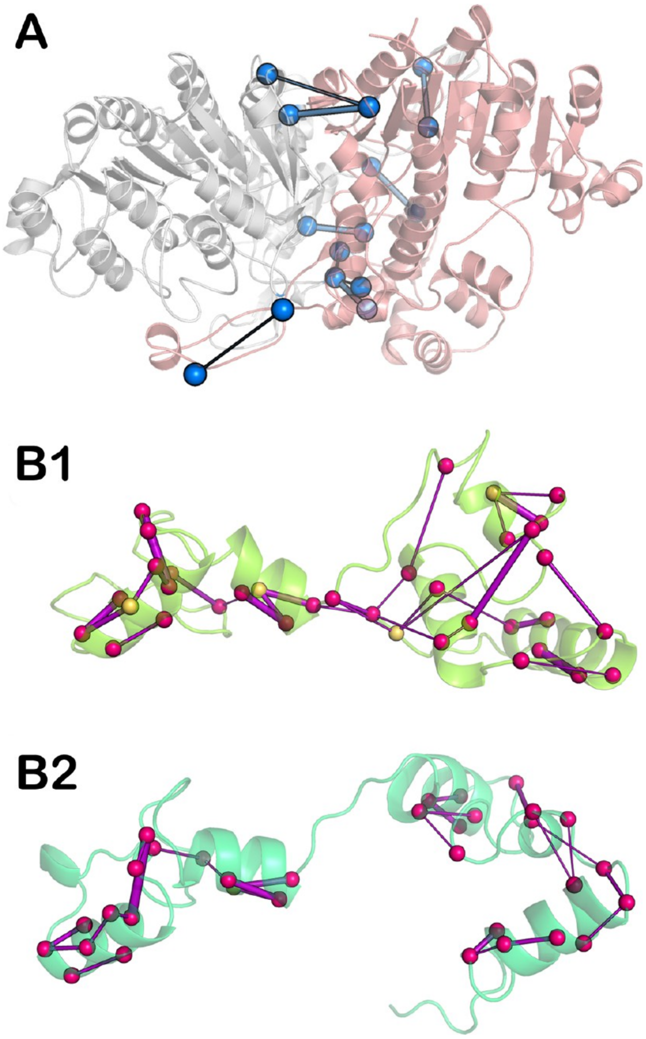


Figure 3. Salt-bridge or H-bond IIN. A) Intermolecular salt bridges in the dimeric cold-adapted *Vibrio sp.* alkaline phosphatase. The sb-IIN for the MD ensemble of VAP was calculated with *PyInteraph* and postprocessed with *xPyder* filtering interactions with persistence lower than 20% and visualizing only the intermolecular salt bridges. The chains A and B of VAP structure are shown in light gray and pink cartoons, respectively. The salt bridges are shown as blue cylinder of thickness proportional to the persistence value, and the residues involved in those salt bridges are shown as spheres centered on the Cα atoms. B) Networks of intramolecular salt-bridge interactions of AT3$_{182-291}$. The sb-IIN for the MD ensemble of AT3$_{182-291}$ was calculated with *PyInteraph* and then postprocessed with *xPyder* filtering out interactions with persistence lower than 20%. The average structures of the two conformational basins identified in the MD simulations are shown as cartoon (1,2), and the Cα atoms of the residues involved in salt bridges are shown as spheres, respectively. The salt bridges are represented as sticks connecting the Cα atoms of residues, and their thickness is proportional to the persistence of the interaction. The Cα of hub residues are highlighted in yellow.

different degree of compactness. A major role for salt bridges organized in networks to stabilize the compact states of AT-3$_{182-291}$ was pointed out by both ESI-MS and simulations. Thus, as an example of application of sb-IIN, we carried out a similar analysis by *PyInteraph* on the MD ensemble of AT-3$_{182-291}$ (Figure 3-B). In particular, the salt bridges were calculated for each of the two subpopulations of AT-3$_{182-291}$. We then estimated the threshold of persistence by the p_{crit} functionality of *filter_graph*, which was 20% in both the cases to discard non-significant and poorly populated interactions in the ensemble. The sb-IIN was then analyzed by *xPyder*[48] tools. In agreement with the previously published data,[51] the results show that AT-3$_{182-291}$ has large networks of salt-bridges, composed by a high number of transient interactions. It turned out that the more compact states (Figure 3-B1) have a higher number of salt-bridge interactions and more interconnected networks compared to the less compact states (Figure 3-B2). Moreover, the analyses identified a higher number of hub residues (i.e., residues connected to more than 3 nodes in the graph) in the more compact states (Figure 3B) that can be related to the propensity for tertiary structures.

As an example of analysis on H-bond networks by *PyInteraph* we here show the case of study of the acylaminoacyl peptidase from the thermophilic organism *Aeropyrum pernix* (ApAAP). ApAAP is a homodimeric serine oligopeptidase composed by a α/β hydrolase fold and a N-terminal β-propeller domain. Its function and structure have been characterized in details by biochemical and crystallographic studies.[52,53] The catalytic triad of ApAAP, composed by Ser445, Asp524, and His556, is structurally stabilized in the closed conformation (Figure 4-B1) of the enzyme by a network of H-bonds involving especially the loops in which are located the catalytic Asp and His residues. In the open conformation (Figure 4-A1), major conformational changes occur in the whole protein. They include a rearrangement of the His loop which becomes completely exposed to the solvent and extremely flexible.[53] To compare the open and closed ApAAp states in a MD framework, we calculated the H-bond network (with a threshold of persistence of H-bond above 30%, as estimated by p_{crit} analysis) by *PyInteraph*. The interaction network was then analyzed by the filtering option of the *xPyder*[48] plugin to map the local H-bonds around the His loop. In particular, the loops harboring Asp524 and His556

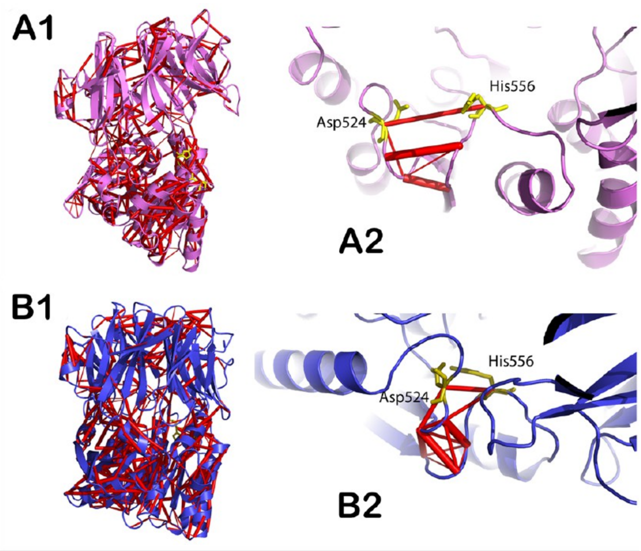


Figure 4. H-bond networks in ApAAP. H-bond networks of two 100 ns MD trajectories of ApAAP starting from a closed (B1) or an open conformer (A1) were calculated with *PyInteraph* and plotted with *xPyder PyMOL* plugin (30% persistence threshold). The thickness of the cylinder is proportional to the H-bond persistence. In the closed conformation (B2) is evident a more persistent H-bond local network between the residues in the loops of the catalytic His and Asp with respect to the open conformation (A2).

showed a clearly different pattern of interactions, with a larger number of more persistent H-bonds connecting the two loops in the closed state compared to the open state (Figure 4-A2, B2), confirming what was observed by the X-ray structures. This difference could also account for the increased flexibility observed for the His loop in the open conformation.

3.2.3. Application-3: Network of Hydrophobic Interactions.

Clusters of hydrophobic interactions are generally found in the protein core, in a buried position, or at the center of interfaces between subunits in a multimeric structure.[69] An example of *PyInteraph* application to decode information from a network of hydrophobic interactions is here presented for the hyperthermophilic ApAAP protein.

The N-terminal α-helix 1 (α1) of ApAAP protrudes from the N-terminal domain, and it connects the β-propeller to the catalytic domain. α1 was demonstrated to play an important role for protein stability since the deletion of the first 21 amino acids of ApAAP affects the temperature-dependence of ApAAP activity.[52] It is also known that those effects are not ascribable to its charged residues, whereas hydrophobic residues seem to play an important role and have been shown to communicate long-range to the catalytic site.[40]

Here, we assessed the capability of a network description of hydrophobic interactions in detecting the paths of long-range communication exerted by the hydrophobic residues in the N-terminal α1 of ApAAP. In particular, we calculated the hydrophobic interaction network by *PyInteraph* and a persistence threshold by *filter_graph* (i.e., 22%). The *graph file* was then analyzed by *graph_analysis* to calculate the shortest paths of long-range communication between each hydrophobic residue of the helix and two hydrophobic residues just in the proximity of the catalytic histidine (H556), i.e. I558 and A554 (Figure 5). In agreement with the previous results, we here showed that the native networks of hydrophobic interactions are compromised and decreased in number upon V16A mutation (Figure 5-A) with respect to the wild type (Figure 5-B). The paths of communication from α1 residues to the proximity of the catalytic histidine (H556) are lost, in particular the paths directed toward A554.

3.2.4. Application-4: To Build a Network of Interactions Based Only on Side-Chain Contacts.

PyInteraph can be also used to build a general network of side-chain contacts, if the default list of residues for hydrophobic interaction is replaced by all the residues (except for glycines). In this case, the calculations will be carried out for each pair of residues within the selected distance cutoff, considering the distance between the center of mass of the two residues on the base of the atomic mass file. An example of this application is reported in Figure 6A for ApAAP, and it will be compared to other networks in the following sections. In particular, Figure 6A-1 illustrates the hubs of the contact graph, as well as an example of paths of long-range communication from each hydrophobic residue of the ApAAP α1 helix and the residues in the proximity of the catalytic histidine (Figure 6-A2). The path search on the contact networks was carried out by the *xPyder*[48] module for Network Analysis upon filtering the map with a persistence cutoff higher than 19%, estimated by the *filter_graph* module.

We compared the results to the description of the paths of communication from the N-terminal helix to the catalytic site of ApAAP achieved by using only the hydrophobic interaction network (Figure 5) or the PSN/DCCM approach.[40] The three approaches are all in overall agreement in identifying paths of long-range communication from the hydrophobic residues in the helix toward the catalytic site or its surroundings, whereas no significant paths can be identified mediated from the charged residues of the helix in the direction of the catalytic site.

3.2.5. Calculation of the Macro-IIN.

PyInteraph has been also designed to provide an unweighted macro-IIN that can be achieved combining all the individual IIN networks of interest upon filtering according to a preselected threshold of persistence. Macro-IIN can be a suitable tool to have an overall view on all the connections between the most persistent interactions in the ensemble. Moreover, hub residues can be analyzed from the macro-IIN graphs, and they can provide a complementary and

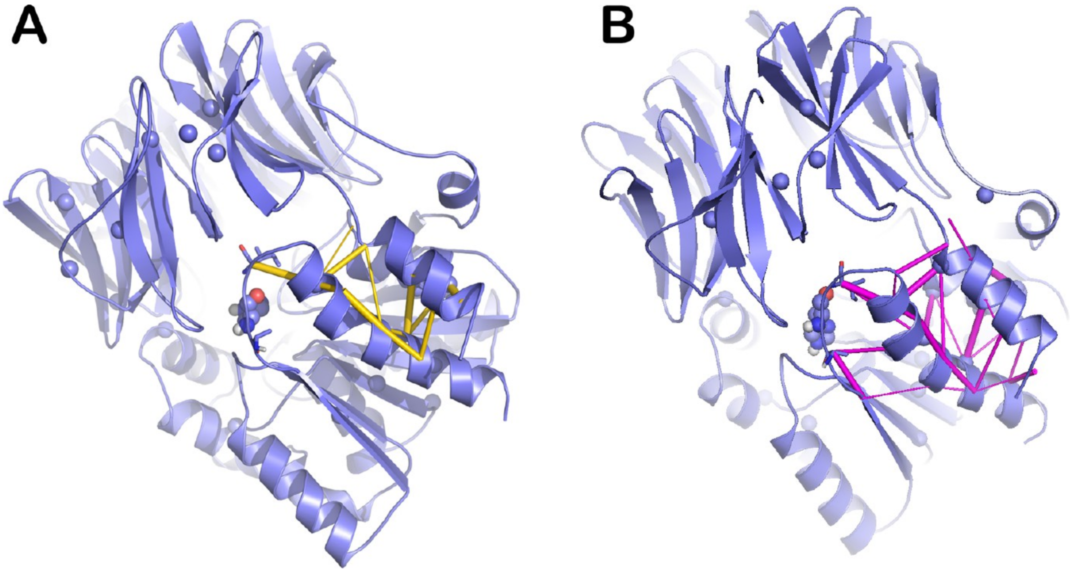


Figure 5. Paths of communication mediated by hydrophobic residues in the N-terminal helix of ApAAP as derived by a graph analysis of the hydrophobic-interaction network in V16A (A) and wt (B) variants. Hubs are shown as spheres, the edges in the paths are indicated by cylinders of thickness proportional to the persistence value of the interaction in the MD ensembles, H549 is shown as sticks and spheres, whereas I551 and A547 are shown as sticks.

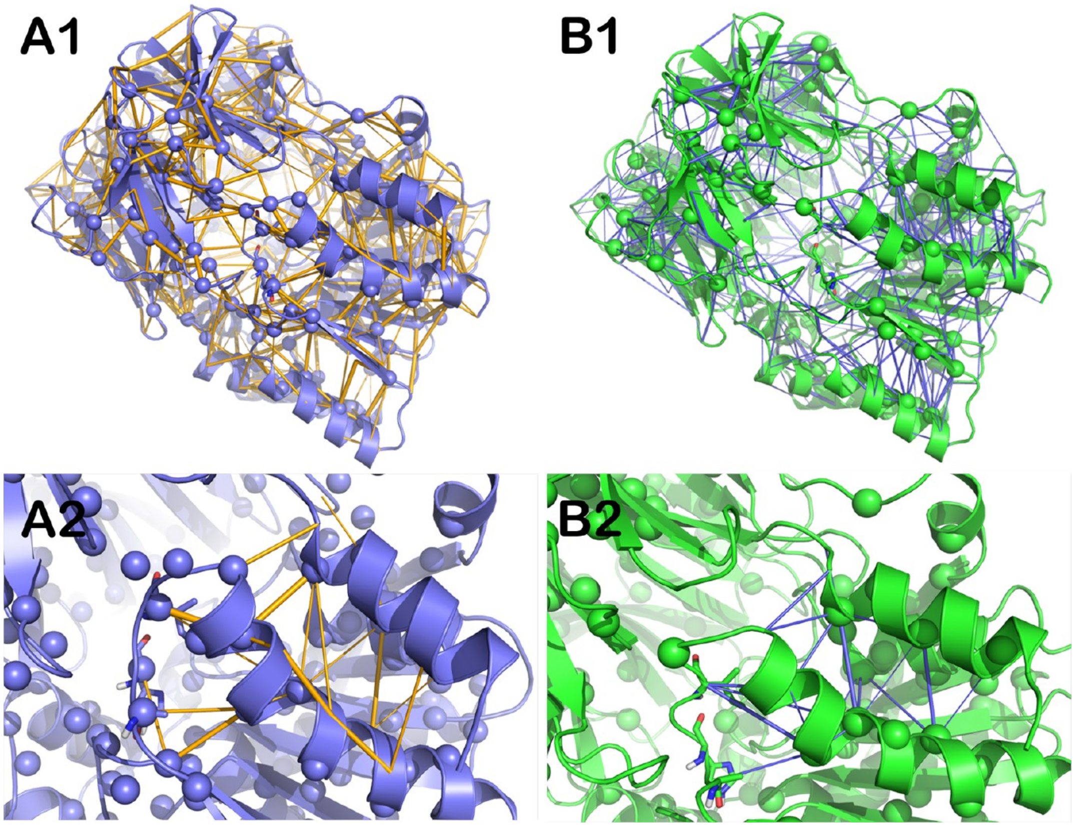


Figure 6. Intramolecular networks as derived by the analysis of side-chain contact maps or energy graphs on ApAAP. A1) The whole network as derived by the analysis of the side-chain contacts with *PyInteraph* and filtering of all the contacts with a persistence higher than with 18%. Contacts are shown as cylinder of thickness proportional to the interaction persistence. The hub residues (i.e., nodes with more than 3 connections in the graph) are shown as spheres centered on the Cα atoms. The catalytic histidine and the two residues in the surrounding I551 and A547 are shown as sticks and spheres. A2) The paths of communication between each hydrophobic residue of ApAAP helix 1 and the I551, A547, and H549 are shown as cylinders as derived by the network analysis performed on the contact map in panel A1. B1) The whole network described by the energy map of ApAAP using the four-distance statistical potential is shown with the same scheme described in Part A. B2) The paths calculated for the same pairs of residues described in panel B1 are shown as detected from the graph analysis of the energy map.

additional source of information to identify crucial residues for protein stability or related to structural communication.

The macro-IIN described in Section 2.3 is, by default, an unweighted graph, it cannot thus be used for network analysis itself. It should be used just to provide an overall view of the location of all the interactions on the protein structure and the reciprocal organization. Nevertheless, the user can supply a weights matrix to the *filter_graph* script. In this case the weights present in the matrix are assigned to the edges encoded in the macro-IIN. Of course, any matrix of the correct shape can be used, thus making the approach general and transferable. For instance, the *filter_graph* tool can be used to combine the macro-IIN with the energy map, so that the pseudoenergy values for each pair of residues can be used as weights for the edges of the macro-IIN and network analysis can be performed on the macro-IIN.

3.2.6. Application-5: Insights on Structural Communication from Interaction Energy Maps. Another functionality of *PyInteraph* is the estimation of interaction energies using a statistical potential based on four-atom distances, called *Hunter*[46] applied to the structural ensemble. This information is likely to complement the ones achieved by the IIN graphs that do not take into account any energetic terms. For comparison we here illustrate the same analyses applied to ApAAP α1 hydrophobic residues using as input only the pseudoenergy map itself and filtering the map for values higher than −0.1 according to the p_{crit} analysis. The map was then postprocessed by *xPyder*[48] to calculate the paths of communication to the catalytic site (Figure 6-B). Interestingly, all the approaches we tried are in agreement in highlighting a long-range communication between the hydrophobic residues of the helix and the proximity of the histidine of the catalytic site. Also most of the paths identified by the pseudoenergy map (Figure 6-B) and the contact map analysis (Figure 6-A) include the same nodes and have the same length, and most of them involved hydrophobic residues, which were the major component of these paths. The same two (contact and pseudoenergy maps) approaches applied to the identification of communication paths from the positively charged residue R18 to the catalytic site cannot identify valid paths in agreement with the fact that this residue when mutated has no effect on the kinetic parameters.[40]

3.2.7. Application-6: Monitoring Interactions Involving Noncanonical Residues or a Nonprotein Ligand. As detailed above, *PyInteraph* supports the introduction of uncommon residue types or molecules through the modification of user-friendly configuration files. For example, residues that have been subject to post-translational modifications or nonprotein ligands can be included. In the following, we show two examples, i.e. the interactions exploited by a phosphoresidue in an E2 enzyme

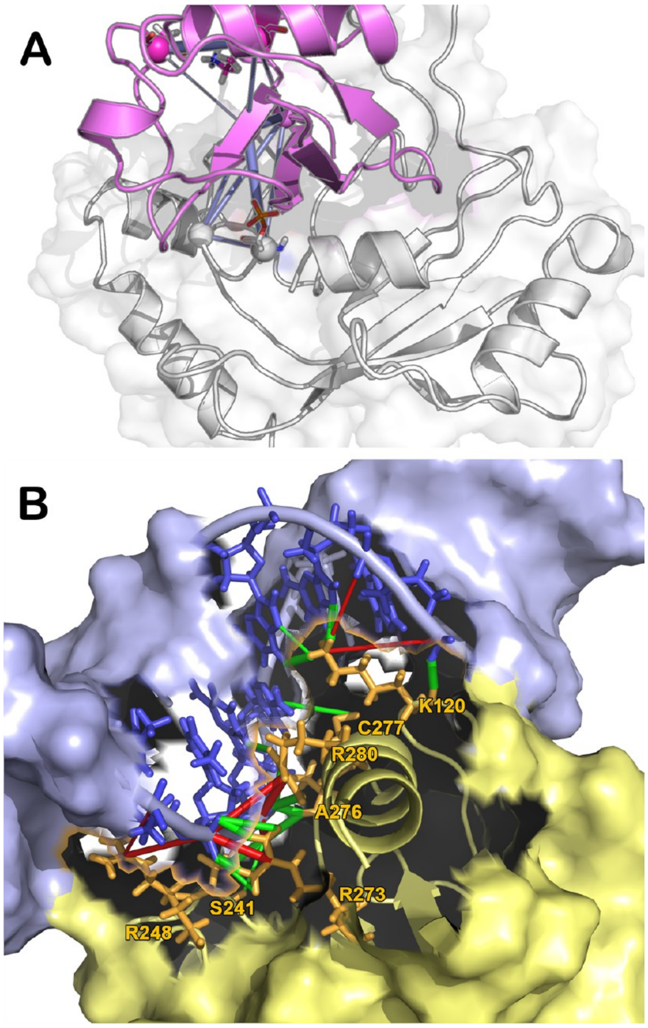


Figure 7. Analysis of intra- and intermolecular interactions including noncanonical residues or nonproteic ligand. A) The role of phospho-S130 of Cdc34 in mediating intermolecular interactions with the Ub molecule is shown. Cdc34 and Ub are shown in light gray and magenta, respectively. The pS130 and the distal sites on Ub (D216 and K191) are shown as sticks. The paths of communication from pS130 to those sites are shown as violet sticks of thickness proportional to the persistence of the interaction in the MD ensemble and residues belonging to the paths that are hubs in the salt-bridge network of the Cdc34-Ub complex are shown as spheres centered on the C-alpha atom. B) Interactions stabilizing the binding between p53 DNA binding domain (p53-DBD) and a p53 responsive element (p53-RE). The p53-DBD and the DNA are shown in yellow and light blue and represented as cartoon and surface. The salt bridges and H-bonds between the protein residues and DNA are shown as red and green cylinders, respectively, with thickness proportional to the persistence of the interaction in MD ensemble. The residues involved in the interactions are highlighted with sticks.

(Figure 7-A) and the interaction between p53 DNA-binding domain and the DNA molecule (Figure 7-B).

Post-translational modifications and phosphorylation in particular have a crucial regulatory and functional role in protein biology. Phosphorylation is a ubiquitous mechanism and the modification of a polar residue, as Ser, Thr, or Tyr in a protein with the addition of a negatively charged group as phosphate can cause large electrostatic perturbations modulating the free energy landscape of the protein and its conformational ensemble.[70] Phosphorylation often results not only in local but also complex and long-range effects often related to switch change in protein function.

As an example of an interaction network involving noncanonical residues, we here presented the long-range paths mediated by a conserved phospho-site in the family 3 of E2 Ub-conjugating enzymes. This phospho-site was demonstrated to be important for the activation of the Ub-charging activity of Cdc34,[71] and it has been also suggested to mediate electrostatic intramolecular interactions with the thiolester-bound Ub molecule.[58] We here analyzed the MD simulation of the phospho-variant of Cdc34 in complex with Ub[58] in terms of a network of salt-bridge interactions with particular attention to the networks that are mediated by the phospho-Ser at position 130. The analysis was carried out modifying the *.ini* configuration file so that the atoms of the phospho group were included as a new "charged group". The final salt-bridge network was analyzed by *filter_graph* to identify the persistence threshold for the analysis (i.e., 20% of persistence). The remaining interactions were analyzed by *graph_analysis* calculating both hubs in the network and the shortest paths of communication from the phospho-S130 residue to other residues of the E2 or of Ub. It turned out that pS130 is a crucial hub of the salt-bridge network, and it is tightly connected not only within the E2 catalytic domain but also mediates several electrostatic intermolecular interactions with the Ub. Indeed, pS130 can reach by multiple paths D216 and K191 of Ub, which are also hub residues in the networks (Figure 7A).

Another example for noncanonical molecules is the analysis of an MD ensemble of p53 DBD in complex with the DNA (Figure 7B). Hundreds of different p53-response elements (p53-REs) have been identified in the human genome.[72] They are found in promoters and enhancers associated with the regulation of genes involved in several cellular pathways such as apoptosis and senescence, and they are selectively activated for transcriptional repression or activation by binding to p53. It has been suggested that subtle differences in p53-REs sequence can trigger variances in the interaction patterns and induce allosteric alterations on p53 DBD, which can, in turn, affect the recruitment of coregulator and the organization of tetrameric p53 in order to activate specific functions.[73] In this context, we performed and analyzed MD simulations of p53 DBD in complex with a canonical p53-REs[59] analyzing salt bridges and H-bonds between the p53 DBD and the DNA by *PyInteraph*. The analysis was performed using a modified version of *charged_groups.ini* and *hydrogen_bonds.ini* configuration files to introduce all the DNA-groups that can be involved in salt bridges or H-bonds with proteins. The p_{crit} was calculated by *filter_graph*, and a cutoff of 20% was identified as a significant threshold of interaction persistence. The interactions were analyzed both by *xPyder*[48] and the *Interaction_Plotter* plugin. Our analyses allow the description of all the intermolecular relevant electrostatic interactions between p53 DBD and DNA, in a MD framework, showing an overall agreement with a previous study.[73] In particular, we identified highly persistent and sequence-specific contacts with the DNA major groove by residues in H2 helix and loop L1 of p53 DBD, as well as contacts with the DNA minor groove by residues in loop L3 (Figure 7B). Moreover, salt bridges between the DNA-backbone phosphate groups and the protein can be identified (Figure 7B). In particular, residues relevant for the interaction with the DNA major groove are K120, C277, and R280. K120 also interacts with the phosphate groups of Gua7 and Gua8. R280 is involved in salt-bridges and H-bonds with phosphates of Gua10' and Thy11'. The interactions with the DNA minor groove are mostly mediated by R248 in loop L3, A276 backbone amide, R273, and S241.

4. CONCLUSIONS

Here we present *PyInteraph*, a novel open-source software designed to calculate intra- and intermolecular interactions in protein structural ensembles, describe them in form of networks of interactions, and perform network analysis on the interaction graphs. It is a versatile tool, and it can accept the structural ensemble in different format and from different sources, i.e. either experimental or simulation-derived ensembles. The program can calculate salt bridges, hydrogen bonds, and hydrophobic interactions, along with their persistence in the structural ensemble. *PyInteraph* also estimates the interaction energy between side chains employing a recently developed knowledge-based potential.[46,47] A graph per each interaction class (intra-intermolecular interaction network, IIN), in which each residue represents a node and the interactions between them the edges, can be computed from the structural ensemble. A tool available in the package, *filter_graph*, can then be used to estimate a significance threshold of persistence for the IINs and to filter them according to this criterion. The identified interaction graphs, one per each type, can be also combined in a comprehensive macro-graph (macro-IIN). The macro-IIN if weighted, for example on the base of the interaction energy, can provide additional insight on the interplay between the different interaction classes and detect paths of structural communication. Plotting of the interactions and of the IINs can be easily performed by plugins for the popular molecular visualization software *PyMOL*. In fact, the software package includes an especially designed *PyMOL* plugin (*interaction_plotter*) to plot the interactions between the individual chemical groups of the residue side chains. Moreover, each IIN, the macro-IIN and the energy graph can be also visualized with the *PyMOL* plugin *xPyder*.[48] The use of straightforward and user-friendly configuration files and flags provides then a great flexibility. Indeed, it permits to extend and enhance the program by including in the analysis modified amino acids and custom molecules to allow the user to deal with more complex cases. The *PyInteraph* package is easy to install and manage under the most common operating systems, as it is based upon open source *Python* libraries.

The program is available free of charge as Open Source software via the GPL v3 license at http://linux.btbs.unimib.it/pyinteraph/.

■ AUTHOR INFORMATION

Corresponding Authors

*E-mail: elena.papaleo@bio.ku.dk, elena.papaleo.78@gmail.com (E.P.).
*E-mail: matteo.tiberti@gmail.com (M.T.).

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This research was supported by CASPUR (*Consorzio Interuniversitario per le Applicazioni di Supercalcolo per Universita' e Ricerca*) Standard HPC Grant 2011 and 2012 to E.P and G.I, as well as by ISCRA-Cineca (projects DynE2-HP10COBO3E, E2-Cons-HP10CQEI1L) to E.P. G.I. is recipient of a Marie Curie Intra European Fellowship. The authors thank Alberto Arrigoni for his involvement in the early stages of the project.

■ REFERENCES

(1) Csermely, P.; Korcsmaros, T.; Kiss, H. J. M.; London, G.; Nussinov, R. Structure and dynamics of molecular networks: a novel paradigm of drug discovery: a comprehensive review. *Pharmacol. Theor.* **2013**, *138*, 333−408.

(2) Böde, C.; Kovács, I. A.; Szalay, M. S.; Palotai, R.; Korcsmáros, T.; Csermely, P. Network analysis of protein dynamics. *FEBS Lett.* **2007**, *581*, 2776−2782.

(3) Vishveshwara, S.; Ghosh, A.; Hansia, P. Intra and inter-molecular communications through protein structure network. *Curr. Protein Pept. Sci.* **2009**, *10*, 146−160.

(4) Atilgan, C.; Okan, O. B.; Atilgan, A. R. Network-based models as tools hinting at nonevident protein functionality. *Annu. Rev. Biophys.* **2012**, *41*, 205−225.

(5) Rahat, O.; Alon, U.; Levy, Y.; Schreiber, G. Understanding hydrogen-bond patterns in proteins using network motifs. *Bioinformatics* **2009**, *25*, 2921−2928.

(6) Vijayabaskar, M. S.; Vishveshwara, S. Interaction energy based protein structure networks. *Biophys. J.* **2010**, *99*, 3704−3715.

(7) Vijayabaskar, M. S.; Vishveshwara, S. Insights into the fold organization of TIM barrel from interaction energy based structure networks. *PLoS Comput. Biol.* **2012**, *8*, e1002505.

(8) Scarabelli, G.; Morra, G.; Colombo, G. Predicting interaction sites from the energetics of isolated proteins: a new approach to epitope mapping. *Biophys. J.* **2010**, *98*, 1966−1975.

(9) Bhattacharyya, M.; Bhat, C. R.; Vishveshwara, S. An automated approah to network features of protein structure ensembles. *Protein Sci.* **2013**, *22*, 1399−1416.

(10) Sengupta, D.; Kundu, S. Role of long- and short-range hydrophobic, hydrophilic and charged residues contact network in protein's structural organization. *BMC Bioinf.* **2012**, *13*, 142.

(11) Aftabuddin, M.; Kundu, S. Hydrophobic, hydrophilic and charged amino acid networks within protein. *Biophys. J.* **2007**, *93*, 225−231.

(12) Atilgan, A. R.; Akan, P.; Baysal, C. Small-world communication of residues and significance for protein dynamics. *Biophys. J.* **2004**, *86*, 85−91.

(13) Vendruscolo, M.; Dokholyan, N.; Paci, E.; Karplus, M. Small-world view of the amino acids that play a key role in protein folding. *Phys. Rev. E* **2002**, *65*, 1−4.

(14) Estrada, E. Universitality in protein residue networks. *Biophys. J.* **2010**, *98*, 890−900.

(15) Ghosh, A.; Vishveshwara, S. A study of communication pathways in methionyl-tRNA synthetase by molecular dynamics simulations and structure network analysis. *Proc. Natl. Acad. Sci. U. S. A.* **2007**, *104*, 15711−15716.

(16) Daily, M. D.; Upadhyaya, T. J.; Gray, J. J. Contact rearrangements form coupled networks from local motions in allosteric proteins. *Proteins* **2008**, *71*, 455−466.

(17) Meireles, L.; Gur, M.; Bakan, A.; Bahar, I. Pre-existing soft modes of motion uniquely defined by native contact topology facilitate ligand binding to proteins. *Protein Sci.* **2011**, *20*, 1645−1658.

(18) Martin, A. J. M.; Vidotto, M.; Boscariol, F.; Di Domenico, T.; Walsh, I.; Tosatto, S. C. RING: networking interacting residues, evolutionary information and energetics in protein structures. *Bioinformatics* **2011**, *27*, 2003−2005.

(19) Doncheva, N. T.; Klein, K.; Domingues, F. S.; Albrecht, M. Analyzing and visualizing residue networks of protein strucutres. *Trends Biochem. Sci.* **2011**, *36*, 179−182.

(20) Vijayabaskar, M. S.; Niranjan, V.; Vishveshwara, S. GraProtStr - Graphs of protein structures: a tool for constructing the graphs and generating graph parameters for protein structures. *Open Bioinf. J.* **2011**, 53−58.

(21) Goncearenco, A.; Mitternacht, S.; Yong, T.; Eisenhaber, B.; Eisenhaber, F.; Berezovsky, I. N. SPACER: server for predicting allosteri communication and effects of regulation. *Nucleic Acids Res.* **2013**, *41*, W266−W272.

(22) Boehr, D. D.; Nussinov, R.; Wright, P. E. The role of dynamic conformational ensembles in biomolecular recognition. *Nat. Chem. Biol.* **2009**, *5*, 789−796.

(23) Vendruscolo, M. Determination of conformationally heterogeneous states of proteins. *Curr. Opin. Struct. Biol.* **2007**, *17*, 15−20.

(24) Barrett, P. J.; Chen, J.; Cho, M.-K.; Kim, J.-H.; Lu, Z.; Mathew, S.; Peng, D.; Song, Y.; Van Horn, W. D.; Zhuang, T.; Sonnichsen, F. D.; Sanders, C. R. The quiet renaissance of protein nuclear magnetic resonance. *Biochemistry* **2013**, 1303−1320.

(25) Manley, G.; Loria, J. P. NMR insights into protein allostery. *Arch. Biochem. Biophys.* **2012**, *519*, 223−231.

(26) Tzeng, S.-R.; Kalodimos, C. G. Protein dynamics and allostery: an NMR view. *Curr. Opin. Struct. Biol.* **2011**, *21*, 62−67.

(27) Brüschweiler, S.; Schanda, P.; Kloiber, K.; Brutscher, B.; Kontaxis, G.; Konrat, R.; Tollinger, M. Direct observation of the dynamic process underlying allosteric signal transmission. *J. Am. Chem. Soc.* **2009**, *131*, 3063−3068.

(28) Klepeis, J. L.; Lindorff-Larsen, K.; Dror, R. O.; Shaw, D. E. Long-timescale molecular dynamics simulations of protein structure and function. *Curr. Opin. Struct. Biol.* **2009**, *19*, 120−127.

(29) Fenwick, R. B.; Esteban-Martín, S.; Salvatella, X. Understanding biomolecular motion, recognition, and allostery by use of conformational ensembles. *Eur. Biophys. J.* **2011**, *40*, 1339−1355.

(30) Esteban-Martín, S.; Bryn Fenwick, R.; Salvatella, X. Synergistic use of NMR and MD simulations to study the structural heterogeneity of proteins. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2012**, *2*, 466−478.

(31) 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*, 6133−6138.

(32) Short, T.; Alzapiedi, L.; Brüschweiler, R.; Snyder, D. A covariance NMR toolbox for MATLAB and OCTAVE. *J. Magn. Reson.* **2011**, *209*, 75−78.

(33) Van den Bedem, H.; Bhabha, G.; Yang, K.; Wright, P. E.; Fraser, J. S. Automated identification of functional dynamics contact networks from X-ray crystallography. *Nat. Methods* **2013**, *10*, 896−902.

(34) Pandini, A.; Fornili, A.; Fraternali, F.; Kleinjung, J. Detection of allosteric signal transmission by information-theoretic analysis of protein dynamics. *FASEB J.* **2012**, *26*, 868−881.

(35) Chiappori, F.; Merelli, I.; Colombo, G.; Milanesi, L.; Morra, G. Molecular mechanism of allosteric communication in Hsp70 revealed by molecular dynamics simulations. *PLoS Comput. Biol.* **2012**, *8*, e1002844.

(36) Laine, E.; Auclair, C.; Tchertanov, L. Allosteric communication across the native and mutated KIT receptor tyrosine kinase. *PLoS Comput. Biol.* **2012**, *8*, e1002661.

(37) Mariani, S.; Dell'Orco, D.; Felline, A.; Raimondi, F.; Fanelli, F. Network and atomistic simulations unveil the structural determinants of mutations linked to retinal diseases. *PLoS Comput. Biol.* **2013**, *9*, e1003207.

(38) Raimondi, F.; Felline, A.; Seeber, M.; Mariani, S.; Fanelli, F. A mixed protein structure network and elastic network model approach to predict the structural communication in biomolecular systems: the PDZ2 domain from tyrosine phosphatase 1E as a case study. *J. Chem. Theory Comput.* **2013**, *9*, 2504−2518.

(39) Blacklock, K.; Verkhivker, G. M. Differential modulation of functional dynamics and allosteric interactions in the Hsp90-cochaperone complexes with p23 and Aha1: a computational study. *PLoS One* **2013**, *8*, e71936.

(40) Papaleo, E.; Renzetti, G.; Tiberti, M. Mechanisms of intramolecular communication in a hyperthermophilic acylaminoacyl peptidase: a molecular dynamics investigation. *PLoS One* **2012**, *7*, e35686.

(41) Papaleo, E.; Lindorff-Larsen, K.; De Gioia, L. Paths of long-range communication in the E2 enzymes of family 3: a molecular dynamics investigation. *Phys. Chem. Chem. Phys.* **2012**, *14*, 12515−12525.

(42) Stacklies, W.; Xia, F.; Gräter, F. Dynamic allostery in the methionine repressor revealed by force distribution analysis. *PLoS Comput. Biol.* **2009**, *5*, e1000574.

(43) Pandini, A.; Fornili, A.; Fraternali, F.; Kleinjung, J. GSATools: analysis of allosteric communication and functional local motions using a structural alphabet. *Bioinformatics* **2013**, *29*, 2053−2055.

(44) Seeber, M.; Felline, A.; Raimondi, F.; Muff, S.; Friedman, R.; Rao, F.; Caflisch, A.; Fanelli, F. Wordom: a user-friendly program for the analysis of molecular structures, trajectories and free energy surfaces. *J. Comput. Chem.* **2011**, *32*, 1183−1194.

(45) Michaud-Agrawal, N.; Denning, E. J.; Woolf, T. B.; Beckstein, O. MDAnalysis: a toolkit for the analysis of molecular dynamics simulations. *J. Comput. Chem.* **2011**, 2319−2327.

(46) Potapov, V.; Cohen, M.; Inbar, Y.; Schreiber, G. Protein structure modelling and evaluation based on a 4-distance description of side-chain interactions. *BMC Bioinf.* **2010**, *11*, 374.

(47) Cohen, M.; Potapov, V.; Schreiber, G. Four distances between pairs of amino acids provide a precise description of their interaction. *PLoS Comput. Biol.* **2009**, *5*, e1000470.

(48) 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**, *279*, 1−6.

(49) Cook, B. W.; Shaw, G. S. Architecture of the catalytic HPN motif is conserved in all E2 conjugation enzymes. *Biochem. J.* **2012**, *445*, 167−174.

(50) Cañadillas, J. M. P.; Tidow, H.; Freund, S. M. V; Rutherford, T. J.; Ang, H. C.; Fersht, A. R. Solution strucutre of p53 core domain: structural basis for its instability. *Proc. Natl. Acad. Sci. U. S. A.* **2006**, *103*, 2109−2114.

(51) Invernizzi, G.; Lambrughi, M.; Regonesi, M. E.; Tortora, P.; Papaleo, E. The conformational ensemble of the disordered and aggregation-protective 182−291 region of ataxin-3. *Biochim. Biophys. Acta* **2013**, *1830*, 5236−5247.

(52) Zhang, Z.; Zheng, B.; Wang, Y.; Chen, Y.; Manco, G.; Feng, Y. The conserved N-terminal helix of acylpeptide hydrolase from archaeon Aeropyrum pernix K1 is important for its hyperthermophilic activity. *Biochim. Biophys. Acta* **2008**, *1784*, 1176−1183.

(53) Harmat, V.; Domokos, K.; Menyhárd, D. K.; Palló, A.; Szeltner, Z.; Szamosi, I.; Beke-Somfai, T.; Náray-Szabó, G.; Polgár, L. Structure and catalysis of acylaminoacyl peptidase: closed and open subunits of a dimer oligopeptidase. *J. Biol. Chem.* **2011**, *286*, 1987−1998.

(54) Ghosh, A.; Brinda, K. V.; Vishveshwara, S. Dynamics of lysozyme structure network: probing the process of unfolding. *Biophys. J.* **2007**, *92*, 2523−2535.

(55) Dokholyan, N. V.; Shakhnovich, B.; Shakhnovich, E. I. Expanding protein universe and its origin from the biological Big Bang. *Proc. Natl. Acad. Sci. U. S. A.* **2002**, *99*, 14132−14136.

(56) Brinda, K. V.; Vishveshwara, S. A network representation of protein structures: implications for protein stability. *Biophys. J.* **2005**, *89*, 4159−4170.

(57) Papaleo, E.; Renzetti, G.; Invernizzi, G.; Asgeirsson, B. Dynamics fingerprint and inherent asymmetric flexibility of a cold-adated homodimeric enzyme. A case study of the Vibrio alkaline phosphatase. *Biochim. Biophys. Acta* **2013**, *1830*, 2970−2980.

(58) Papaleo, E.; Casiraghi, N.; Arrigoni, A.; Vanoni, M.; Coccetti, P.; De Gioia, L. Loop 7 of E2 enzymes: an ancestral conserved functional motif involved in the E2-mediated steps of the ubiquitination cascade. *PLoS One* **2012**, *7*, e40786.

(59) Cho, Y.; Gorina, S.; Jeffrey, P. D.; Pavletich, N. P. Crystal structure of a p53 tumor suppressor-DNA complex: understanding tumorigenic mutations. *Science* **1994**, *265*, 346−355.

(60) Hess, B.; Bekker, H.; Berendsen, H.; Fraaije, J. LINCS: a linear constraint solver for molecular simulations. *J. Comput. Chem.* **1997**, *18*, 1463−1472.

(61) Darden, T.; York, D.; Pedersen, L. Particle mesh Ewald: an N log (N) method for Ewald sums in large systems. *J. Chem. Phys.* **1993**, *98*, 10089−10092.

(62) Munson, M.; Balasubramanian, S.; Fleming, K. G.; Nagi, A. D.; O'Brien, R.; Sturtevant, J. M.; Regan, L. What makes a protein a protein? Hydrophobic core designs that specify stability and structural properties. *Protein Sci.* **1996**, *5*, 1584−1593.

(63) Bullock, A. N.; Fersht, A. R. Rescuing the function of mutant p53. *Nat. Rev. Cancer* **2001**, *1*, 68−76.

(64) Ye, Y.; Rape, M. Building ubiquitin chains: E2 enzymes at work. *Nat. Rev. Mol. Cell Biol.* **2009**, *10*, 755−764.

(65) Kumar, S.; Nussinov, R. Relationship between ion pair geometries and electrostatic strenghts in proteins. *Biophys. J.* **2002**, *83*, 1595−1612.

(66) Kumar, S.; Nussinov, R. Close-range electrostatic interactions in proteins. *ChemBioChem* **2002**, *3*, 604−617.

(67) Siddiqui, K. S.; Cavicchioli, R. Cold-adapted enzymes. *Annu. Rev. Biochem.* **2006**, *75*, 403−433.

(68) Dyson, H. J.; Wright, P. E. Intrinsically unstructured proteins and their functions. *Nat. Rev. Mol. Cell Biol.* **2005**, *6*, 197−208.

(69) Tsai, C. J.; Lin, S. L.; Wolfson, H. J.; Nussinov, R. Studies of protein-protein interfaces: a statistical analysis of the hydrophobic effect. *Protein Sci.* **1997**, *6*, 53−64.

(70) Narayanan, A.; Jacobson, M. P. Computational studies of protein regulation by post-translational phosphorylation. *Curr. Opin. Struct. Biol.* **2009**, *19*, 156−163.

(71) Coccetti, P.; Tripodi, F.; Tedeschi, G.; Nonnis, S.; Marin, O.; Fantinato, S.; Cirulli, C.; Vanoni, M.; Alberghina, L. The CK2 phosphorylation of catalytic domain of Cdc34 modulates its activity at the G1 to S transition in Saccharomyces cerevisiae. *Cell Cycle* **2008**, *7*, 1391−1401.

(72) Wei, C.-L.; Wu, Q.; Vega, V. B.; Chiu, K. P.; Ng, P.; Zhang, T.; Shahab, A.; Yong, H. C.; Fu, Y.; Weng, Z.; Liu, J.; Zhao, X. D.; Chew, J.-L.; Lee, Y. L.; Kuznetsov, V. A.; Sung, W.-K.; Miller, L. D.; Lim, B.; Liu, E. T.; Yu, Q.; Ng, H.-H.; Ruan, Y. A global map of p53 transcription-factor binding sites in the human genome. *Cell* **2006**, *124*, 207−219.

(73) Pan, Y.; Nussinov, R. Lysine 120 interactions with p53 response elements can allosterically direct p53 organization. *PLoS Comput. Biol.* **2010**, *6*, e1000878.