**II. Theory**

Graph theory is the study of mathematical structures, called graphs, consisting of nodes and edges used to model pairwise relations between objects. The origins of graph theory date back to the eighteenth century, when Euler solving the dilemma of the Seven Bridges of Königsberg. Since then, network models have been widely applied to questions in a wide range of fields, including the physical and biological sciences. Here, we focus on graphs of proteins where the nodes correspond to amino acid residues and the edges are defined according to the strength of pairwise correlations established among interacting residues.[48](https://paperpile.com/c/TZ99ZK/w6bm) Protein networks have been traditionally generated using a variety of models, including protein correlation networks (PCNs),[2](https://paperpile.com/c/TZ99ZK/fT4T) protein structure networks,[49](https://paperpile.com/c/TZ99ZK/py2F) protein contact networks,[50](https://paperpile.com/c/TZ99ZK/UFUU) residue interaction graphs,[9,51](https://paperpile.com/c/TZ99ZK/7EER+psM4) and residue networks.[52](https://paperpile.com/c/TZ99ZK/mRE8) In PCNs, the network is constructed from the correlation of a time-dependent variables from MD simulations. The entries of the correlation matrix become the edges of undirected networks, with the nodes corresponding to the elements for which the correlation is computed.

**II.1 Node-level descriptors**

We focus on allosteric communication in proteins described by correlation networks obtained from the analysis of correlations of atomic displacements, torsion angles, and electrostatic energies observed in MD simulations. The analysis of correlations is based on *F* snapshots of configurations (frames) sampled from MD simulations at equidistant time steps.

*Atomic Displacements*

Cartesian atomic displacements of residue *i* are computed for each frame of the MD simulation, as follows:

, [1]

resulting in a matrix of size *F* × 3*N*, called the *atomic displacement matrix 1* (*adm1*), with *N* the number of atoms included in the correlation analysis and ***qi****= {xi, yi, zi,}* the coordinates of each node *i* (e.g., alpha-carbon of each amino acid residue). The displacement magnitudes are computed, as follows:

, [2]

resulting in a matrix of size *F* × *N*, called *adm2,* where each element corresponds to the atomic displacement of a given amino acid residue for each frame of the MD simulation.

*Torsion Angles*

Four torsional coordinates, including sine and cosine backbone torsions and dihedrals, are recorded for each amino acid residue along the MD simulation in the matrix *tam1* of size *F* × 4*N*. Each matrix entry corresponds to the sine/cosine transformed / backbone dihedral for a given residue and snapshot of the MD simulation,

. [3]

The magnitude of the dihedral displacement is computed, as follows:

, [4]

resulting in a matrix of size *F\*N* called *tam2*.

*Electrostatic Energies*

The analysis of correlations of electrostatic energies is based on one descriptor per amino acid residue, corresponding to either the hydrogen bond donor energy, hydrogen bond acceptor energy, or the sum of the donor and acceptor energies, according to the Kabsch-Sander formalism.[53](https://paperpile.com/c/TZ99ZK/Uf04) We focus on electrostatic interactions between the CO and NH backbone groups for each frame of the MD simulation, as follows:

, [5]

resulting in a 3-mode tensor of size *F* × *N* × *N*, where rows correspond to hydrogen bond acceptor (CO) groups, and columns correspond to hydrogen bond donor (NH) groups. From this matrix, we can sum across rows to obtain the hydrogen bond donor energy of each residue, resulting in the *Kabsch-Sander donor matrix* (*ksdm*) of size *F* × *N*, where each element is the hydrogen bond donor energy for a given residue and frame of the MD simulation. Analogously, summing across columns yields the *ksam* matrix corresponding with the acceptor energy for a given residue and frame of the MD simulation. Additionally, we can sum the donor and acceptor energies for each residue, resulting in the matrix *ksdam* of size *F* × *N*, where each element is the sum of hydrogen bond donor and acceptor energies for each residue and frame of the MD simulation.

**II.2 Pairwise correlations**

*MdiGest* allows for calculations of residue-residue couplings using four possible measures of correlation. Dynamic cross correlation, Pearson correlation, generalized correlation based on the linearized mutual information, and generalized correlation obtained from the mutual information are computed from each of the three descriptors of amino acid residues, including atomic displacements, torsion angles, and electrostatic energies.

*Dynamic cross-correlation and Pearson correlation*

The dynamic cross-correlations (*dcc*) between residues *x* and *y* are computed for the *F* frames of an MD simulation, according to the following equation:

. [6]

The *dcc* matrices are of size *F* × c*N*, where c is either 3 (for atomic coordinates), 4 (for torsion angles), or 1 (for electrostatic energies) as provided by matrices *adm1*, *tam1* and *ksdam*, resulting in matrices of size 3*N* × 3*N* and 4*N* × 4*N*, respectively, which are then averaged to obtain an *N* × *N* pairwise *dcc* coefficient.

We compute Pearson correlation coefficients (*pcc*) from *adm2*, *tam2*, *ksdam*, as follows:

, [7]

where each element is the respective pairwise correlation coefficient.

*Generalized correlation*

Generalized correlation coefficients (*gcc*) are computed from either the linearized mutual information, or the mutual information, as follows:

. [8]

Using a Gaussian estimator one can approximate the (linearized) mutual information[2](https://paperpile.com/c/TZ99ZK/fT4T) between two variables, as follows:

, [9]

where the covariances are computed from the matrices *adm1*, *tdm1* or *ksdm*, *ksam*, *ksdam* for each pair of descriptors *x*, *y*, as follows:

. [10]

The resulting matrix accounts for only linear correlations, thus the corresponding correlation matrix is referred as *gcc-lmi*.

Additionally, we can compute *gcc-mi* coefficients from the mutual information,

, [11]

that also accounts for nonlinear correlations. Following the derivation by Kraskov et al,[54](https://paperpile.com/c/TZ99ZK/d2kp) we approximate the mutual information, as follows:

, [12]

where *k* is a parameter defining the number of nearest neighbors, *F* is the total number of frames in the MD simulation, and and are the numbers of frames in which the positions of nodes *x* and *y* are within a specified distance cutoff. The digamma function introduced by Eq. [12] is defined, as follows:

, [13]

where *j = k, F, nx, ny*. Furthermore, is the ensemble average of the sum of digamma functions applied to and , where and are varied for each calculation of according to a distance cutoff, including k-nearest neighbors for each node *x* and *y* in each frame along the MD simulation.

In the implementation of the method, *MdiGest* first initializes an *N* × *N*, array with each entry corresponding to , computed from *k* and *F*. To solve for , we first define **x** to be an *F* × *f* matrix where *f* is 3 for atomic displacements, 4 for torsion angles, and 1 for electrostatic energies for each node *x*. We define **y** analogously for node *y.* The concatenated [**x,y**] array of size 2*F* × *f* is fed into a nested KDTree object, utilizing the Chebyshev distance metric, i.e. the maximum absolute distance in one dimension of two *n*-dimensional points. The KDTree provides the distances to the *k*-nearest neighbors for each point. An additional KDTree object, for each node (*x* and *y*, individually) is used to compute the digamma for the points in each of these two trees, at all distance cutoffs defined by the outer KDTree. Once the average is computed , the mutual information between nodes *x* and *y* is readily obtained.

**II.3 Eigenvector Centrality**

The eigenvector centrality measures the relative importance of nodes in establishing correlations in the network.[55–57](https://paperpile.com/c/TZ99ZK/sEJs+HIoI+SSth) From each set of pairwise correlation coefficients, we can construct a corresponding adjacency matrix and obtain the eigenvector centralities of the nodes in the network as established by that specific correlation coefficient.

*MdiGest* allows for construction if adjacency matrices, A, of size *N* × *N*, for each set of correlation coefficients (*dcc*, *pcc*, *gcc-lmi*, *gcc-mi*) based on one of three atomic descriptors (atomic displacements, torsion angles, electrostatic energies). Each entry of A is a pairwise correlation coefficient representing an edge between two nodes in the network. From the adjacency matrix, the eigenvector centrality is computed, as follows:

.[14]

According to the Perron-Frobenius theorem, the entries of the eigenvector corresponding to the largest eigenvalue can be defined to be all positive real numbers, defining the centrality values for each node (residue) in the network (protein). Each entry of the eigenvector **c** with maximum eigenvalue thus corresponds to the eigenvector centrality of the given node. The eigenvector centrality values therefore quantify the importance of each node in the eigenvector of maximum eigenvalue, measuring how much each node of the protein network contributes to the correlation in the network.

**II.4 Networks obtained with MDiGest**

*MDiGest* allows to build a variety of protein correlation networks for a comprehensive analysis of allostery in the system. As described in the previous section, correlations can be quantified by the Pearson correlation, the generalized correlation from linearized mutual information, and the generalized correlation from mutual information using various different node-level descriptors, including atomic displacements, torsion angles, and electrostatic energies. As recently shown,[29](https://paperpile.com/c/TZ99ZK/92NB) the electrostatic eigenvector centrality (EEC) measure yields improved correspondence with the experimental analysis of allostery based on NMR data. The information provided by EEC generally resembles that obtained from correlations of backbone dihedrals, confirming that correlated electrostatic couplings account for both localized motions and overall conformational changes.[21](https://paperpile.com/c/TZ99ZK/tmNP) Furthermore, the decomposition between donor and acceptor energies provides an additional layer for an in-depth interpretation of the underlying dynamics.