Elucidating Partial Folding State of Bovine Pancreatic Trypsin Inhibitor by a Combined Study of Molecular Dynamics Simulations, Information Theory, Molecular Graph Theory, and Machine Learning

Hiqmet Kamberaj*

International Balkan University, Department of Computer Engineering, Makedonsko-Kosovska Brigada BB, Skopje, R of North Macedonia

Abstract

Using a notably large amount of data in investigating physical and chemical phenomena demands new statistical and computational approaches; besides, the cross-validations require well-established theoretical frameworks. This study aims to validate the statistical efficiency of alternative definitions for the information-theoretic measures, such as transfer entropy, using the so-called (α, q)-framework. The primary goal is to find measurements of high-order correlations that preserve information-theoretic properties of information transfer between the components of a dynamical system (such as a protein) due to local operations. Besides, this study aims to decode the information contained in the amino acid sequence establishing a three-dimensional protein structure by comparing the amino acids physical-chemical properties with their ranked role in the protein interaction network topology using new graph-theoretic measures based on the constructed digraph models of (α, q) information transfer within a heat flow kernel embedding framework. Moreover, this study aims to use the Deep Graph Convolution Neural Networks for classifying the role of each

*Corresponding author
Email address: h.kamberaj@gmail.com (Hiqmet Kamberaj)

amino acid in a protein trained upon short equilibrium structure fluctuations at sub-nanosecond time scales.

In particular, this study examines the influence of disulphide bridges on the three-dimensional structure of the Bovine Pancreatic Trypsin Inhibitor wild type and mutated analogue protein.
Keywords: (α, q) Transfer Entropy, Heat Kernel Molecular DiGraph Embedding, Heat and Information Flow, Deep Graph Convolution Neural Networks, Gibbs Free Energy

---

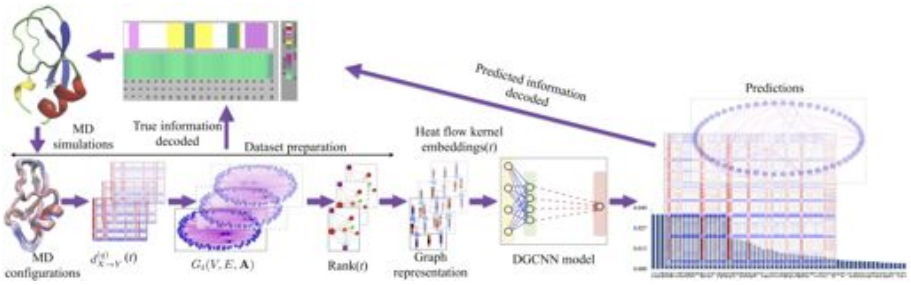

Figure 1: Graphical abstract.

1. Introduction

Although slow conformational motions of large biological molecules studies are limited in time and size scales, atomistic models (AMs) combined with the molecular dynamics (MD) technique are widely used to understand the internal atomic motion of biomolecular systems [1, 2, 3]. Furthermore, a considerably large amount of the data collected to investigate physical and chemical phenomena will require new statistical and computational approaches to be significantly efficient. Moreover, the cross-validation of these new computational approaches requires vast data collected from the simulations and well-established theoretical frameworks.

The problem of encoding the protein structure into amino acid sequences remains unsolved. Information contained in the amino acid sequences can be classified as overlapping of physical-chemical properties of amino acids. The information-theoretic measures, such as mutual information [4], have already been considered [5] to be sensitive to linear and nonlinear statistical correlations, and an improvement compared to Pearson correlation measures [6], supported by some previous studies [7, 8]. These measures can measure the statistical mutual information content between the amino acids of a protein, considered components of an interaction network system, by focusing on the position fluctuations of a group of atoms. On the other hand, correlation coefficients (such as mutual information or Pearson correlation coefficient) are symmetric measures; that is, from the measured mutual information content, one can not distinguish the *sender* (or *source*) and *receiver* (or *sink*) amino acids of the information flow in the interaction network of amino acid sequences [8]. Therefore, the so-called *transfer entropy* [9, 10, 11, 12, 13] showed as an information-theoretic measure of the information flow. The transfer entropy is also sensitive to nonlinear statistical dependences. It enables, as an asymmetric measure, measuring the directional information flow between two amino acids in a protein interaction network and distinguishing between the source amino acid (that is, the driver

amino acid of the interaction network pathway signal) and the sink amino acid (that is, the signal receiver) [8]. The transfer entropy has successfully been used in studying signal pathway sampling in biological system [14], altering the information flow in the amino acid interaction network due to mutation for studying partial protein unfolding [8], changes in the information flow pathways in proteins due to ligand binding [15], studying the hinge influence on the LAC repressor headpiece, information flow in the protein allostery [16, 17], studying allosteric communication landscapes in proteins [18, 19], and studying the heat flow random walks at the protein-RNA interface [20].

The statistical analysis frameworks have been established [20] (and the references therein) to study the amino acids interaction network associated with the topology of the information flow in proteins. The weighted directed graph model was chosen to describe mathematically the topological network of the anisotropic diffusion of the heat flow in proteins. Furthermore, the information-theoretic measures, such as Shannon formalism of transfer entropy [9, 8], are used to describe the geometrical properties of the graphs, defined by the equilibrium structure fluctuations at sub-nanoseconds time scales from molecular dynamics simulations.

In this study, we aim to cross-validate the statistical efficiency of newly developed alternative definitions for the information-theoretic measures of the transfer entropy between time series, based on the so-called (α, q)-framework introduced in Ref. [21]. The primary goal is to find measurements of high-order correlations that satisfy information-theoretic properties of information transfer, mainly due to local operations [22]. Besides, they should be easy to compute using the existing algorithms [8, 23]. Furthermore, this study aims to decode the information contained in the amino acid sequence that establishes a protein structure by comparing the physical-chemical properties of amino acids with their role in the protein interaction network topology; for that, new graph-theoretic measures are introduced based on the constructed digraph models of

(α, q) information transfer between amino acids [20]. The primary aim is to develop a data-driven approach based on causal reconstruction and graph network measures on the causal effects [24] that will identify amino acids acting as major system structure drivers and gateways (or hubs) of equilibrium structure fluctuations at a given time scale from MD simulations.

Moreover, this study aims to use the so-called Deep Graph Convolution Neural Networks (DGCNN) [25], using a new algorithm developed in this study, for classifying the amino acid role in a protein structure trained upon short equilibrium structure fluctuations at sub-nanoseconds time scales from molecular dynamics simulations. In this study, a heat kernel embedding framework [26, 27] is employed to encapsulate information flow in a DiGraph using eigenvalue decomposition of the Laplacian to compute the embedding coordinates matrix of every vertex in the graph.

In particular, this study aims to examine the influence of disulphide bridges on the three-dimensional (3D) structure of the Bovine Pancreatic Trypsin Inhibitor (BPTI) protein (wild type and its mutated analogue) [28, 29].

2. The (α, q)-Framework of Information Flow

In the Supporting Information materials associated with this study, we introduce the so-called (α, q)-framework, according to Ref. [21]. In particular, we describe the (α, q)-definitions of the entropy and mutual information for the discrete probability distribution Markovian time processes, aiming at maximising the mutual information and hence the channel capacity. Then, we have extended formalism for the continuous probability distribution Markovian time processes, by defining the Sharma-Mittal [30], Rényi [31], and Tsallis [32] differential entropies and corresponding mutual information. Besides, we cross-validate the statistical efficiency of (α, q)-framework by comparing it with Shannon formalism for different values of α and q; in this study, we consider two different

intuitive linearly and non-linearly coupled Markovian processes describing the input and output of a Gaussian-like noisy channel and XOR Bernoulli-like noisy channel, respectively. Moreover, we describe the numerical analysis of two newly developed algorithms for computation of the information-theoretic measures, based on the so-called symbolic encoding and kernel density estimators using artificial intelligence techniques provided in Python programming language paradigm within *sklearn* package. Note that the possibilities for extending the Shannon framework of the information theory have already been attempted [33], relying on parametrisation of the information content by introducing the non-additivity property in uncertainty. The advantage is that these new frameworks allow developing higher-order measures sensitive to the non-linear couplings and noisy signals in dynamical systems.

In the following, we introduce a new theoretical framework for the transfer entropy, the so-called (α, q) transfer entropy.

2.1. The (α, q) Transfer Entropy

The transfer entropy, $T_{Y\to X}$, is defined as [9]:

$$T_{Y\to X} = \sum_k p(x_{k+\delta}, \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}) \log \frac{p(x_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y})}{p(x_{k+\delta} \mid \mathbf{x}_k^{\mu_x})} \tag{1}$$

where k is a time index and the information is in units of *nats*. The vectors $\mathbf{x}_k^{\mu_x}$ and $\mathbf{y}_k^{\mu_y}$ are the state vectors of the Markovian processes and $\mu_{x/y} \equiv (m_{x/y}, \tau_{x/y})$ are the embedded parameters, as described in Supporting information. Similarly, the transfer entropy, $T_{X\to Y}$, is given as:

$$T_{X\to Y} = \sum_k p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}, y_{k+\delta}) \log \frac{p(y_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y})}{p(y_{k+\delta} \mid \mathbf{y}_k^{\mu_y})} \tag{2}$$

This formulation of the transfer entropy represents a dynamical measure, as a generalisation of the entropy rate to more than one element to form a mutual information rate [9]. In Eq. 1, $p(x_{k+\delta}, \mathbf{x}^{\mu_x}, \mathbf{y}_k^{\mu_y})$ is the joint probability distribution of observing future value $x_{k+\delta}$ and the pasts of $\mathbf{x}_k^{\mu_x}$ and $\mathbf{y}_k^{\mu_y}$,

$p(x_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y})$ is the conditional probability of observing the future value of X given the past values of both X and Y, and $p(x_{k+\delta} \mid \mathbf{x}_k^{\mu_x})$ is the conditional probability of observing the future of X knowing its past. From Eq. 1, the transfer entropy can also be expressed in terms of Shannon entropies:

$$\begin{aligned} T_{Y\to X} &= H(\mathbf{X}^{\mu_x+1}) - H(\mathbf{X}^{\mu_x}) - H(\mathbf{X}^{\mu_x+1}, \mathbf{Y}^{\mu_y}) + H(\mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y}) \\ &\equiv I\left(X_{+\delta}; \mathbf{Y}^{\mu_y} \mid \mathbf{X}^{\mu_x}\right) = H(X_{+\delta} \mid \mathbf{X}^{\mu_x}) - H(X_{+\delta} \mid \mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y}) \end{aligned} \tag{3}$$

where $I\left(X_{+\delta}; \mathbf{Y}^{\mu_y} \mid \mathbf{X}^{\mu_x}\right)$ is the conditional mutual information [11, 8, 34]. Equation 3 indicates that the transfer entropy, $T_{Y\to X}$, can be interpreted as the average amount of information contained in the source about next state $X_{+\delta}$ of the variable X that was not already contained in the past of X.

Similarly, $T_{X\to Y}$ is given as

$$\begin{aligned} T_{X\to Y} &= H(\mathbf{Y}^{\mu_y+1}) - H(\mathbf{Y}^{\mu_y}) - H(\mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y+1}) + H(\mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y}) \\ &\equiv I\left(Y_{+\delta}; \mathbf{X}^{\mu_x} \mid \mathbf{Y}^{\mu_y}\right) = H(Y_{+\delta} \mid \mathbf{Y}^{\mu_y}) - H(Y_{+\delta} \mid \mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y}) \end{aligned} \tag{4}$$

In this study, the statistical efficiency of alternative definitions of the transfer entropy measures between time series is validated, based on (α, q)-framework [21], as described in Supporting Information. Note that the volume of an d-dimensional space is such that $Vol(\Omega_\epsilon^{(d)}) \approx 2^{dH(X)}$, which is the volume of the smallest set $\Omega_\epsilon^{(d)}$ that contains most of probability distribution for a random process X [4]. That is, in d-dimensional, small Shannon entropy values imply that the random process is confined to a small effective volume while large entropy values imply that the random process is confined to a large effective volume. By definition [4], the set $\Omega_\epsilon^{(d)}$ is the smallest volume set with probability $P(\Omega_\epsilon^{(d)}) \geq 1 - \epsilon$ to the first order in the exponent, where ϵ is an infinitesimally small number. That corresponds to Gaussian-like Markovian processes, which are characterised by short tails, and hence these measures do not maximise the mutual information or channel capacity, in particular, for noisy channels. In contrary, (α, q)-information measures can be used to count for noisy random processes characterised by

longer tails. In particular, the primary goal is to find measurements of high-order correlations that satisfy information-theoretic properties of information transfer, mainly due to local operations [22], and being easy to compute using the existing algorithms [8, 23].

In this study, we define new measures of the transfer entropies, the so-called here (α, q) transfer entropies, as:

$$\begin{aligned} T^{(\alpha,q)}_{Y\to X} &= I_{(\alpha,q)}\left(X_{+\delta}; \mathbf{Y}^{\mu_y} \mid \mathbf{X}^{\mu_x}\right) \\ &= H_{(\alpha,q)}(X_{+\delta} \mid \mathbf{X}^{\mu_x}) \ominus_q H_{(\alpha,q)}(X_{+\delta} \mid \mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y}) \\ &= \left[H(\mathbf{X}^{\mu_x+1}) \ominus_q H(\mathbf{X}^{\mu_x})\right] \\ &\quad \ominus_q \left[H_{(\alpha,q)}(\mathbf{X}^{\mu_x+1}, \mathbf{Y}^{\mu_y}) \ominus_q H_{(\alpha,q)}(\mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y})\right] \end{aligned} \tag{5}$$

Similarly,

$$\begin{aligned} T^{(\alpha,q)}_{X\to Y} &= I_{(\alpha,q)}\left(Y_{+\delta}; \mathbf{X}^{\mu_x} \mid \mathbf{Y}^{\mu_y}\right) \\ &= H_{(\alpha,q)}(Y_{+\delta} \mid \mathbf{Y}^{\mu_y}) \ominus_q H_{(\alpha,q)}(Y_{+\delta} \mid \mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y}) \\ &= \left[H_{(\alpha,q)}(\mathbf{Y}^{\mu_y+1}) \ominus_q H_{(\alpha,q)}(\mathbf{Y}^{\mu_y})\right] \\ &\quad \ominus_q \left[H_{(\alpha,q)}(\mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y+1}) \ominus_q H_{(\alpha,q)}(\mathbf{X}^{\mu_x}, \mathbf{Y}^{\mu_y})\right] \end{aligned} \tag{6}$$

Here, the function $x \ominus_q y$ is defined as [21]:

$$x \ominus_q y = \frac{x - y}{1 + (1-q)y} \tag{7}$$

which reduces to standard difference for $q = 1$. In Eq. 5 (or Eq. 6), in general, $H_{(\alpha,q)}(X)$ is the Sharma-Mittal entropy [30]:

$$H_{(\alpha,q)}(X) = \eta_q\left(H_\alpha(X)\right) \tag{8}$$

Here, $H_\alpha(X)$ is Rényi entropy (see Eq. SI2 in Supporting Information) and η_q is the function [21]:

$$\eta_q(x) = \frac{\exp\left((1-q)x\right) - 1}{1-q} \tag{9}$$

Note that for $q = 1$ and $\alpha \neq 1$, equations 5 and 6 reduce to Rényi transfer entropy formulations [35], based on α-framework of entropy while for $q \neq 1$ and

$\alpha = 1$, they reduce to the so-called here Tsallis transfer entropies, based on the so-called here q-framework. When $q = \alpha = 1$, equations 5 and 6 give the standard transfer entropies based in the Shannon entropy formulation [9].

2.2. Tsallis Local Transfer Entropy

The so-called local Shannon transfer entropy measure is defined as [36]:

$$t_{Y \to X}(k+\delta) = h(x_{k+\delta} \mid \mathbf{x}_k^{\mu_x}) - h(x_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}) \tag{10}$$

Note that everywhere mentioned in this study, δ denotes a time parameter, which indicates an average time-length correlation between X and Y (typically, $\delta = 1$). In the right-hand side of Eq. 10, the local condition entropies are defined as:

$$\begin{aligned} h(x_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}) &= -\Big(\log p(\mathbf{x}_k^{\mu_x+1}, \mathbf{y}_k^{\mu_y}) - \log p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y})\Big) \\ h(x_{k+\delta} \mid \mathbf{x}_k^{\mu_x}) &= -\Big(\log p(\mathbf{x}_k^{\mu_x+1}) - \log p(\mathbf{x}_k^{\mu_x})\Big) \end{aligned} \tag{11}$$

Similarly, the local transfer entropy, $t_{X \to Y}(k+\delta)$, is defined as:

$$t_{X \to Y}(k+\delta) = h(y_{k+\delta} \mid \mathbf{y}_k^{\mu_x}) - h(y_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}) \tag{12}$$

where

$$\begin{aligned} h(y_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}) &= -\Big(\log p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y+1}) - \log p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y})\Big) \\ h(y_{k+\delta} \mid \mathbf{y}_k^{\mu_x}) &= -\Big(\log p(\mathbf{y}_k^{\mu_y+1}) - \log p(\mathbf{y}_k^{\mu_y})\Big) \end{aligned} \tag{13}$$

In this study, we introduce a new measure of the local transfer entropy, named as *Tsallis local transfer entropy*, based on the Tsallis q-framework:

$$t^{(q)}_{X \to Y}(k+\delta) = h_q(y_{k+\delta} \mid \mathbf{y}_k^{\mu_x}) - h_q(y_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}) \tag{14}$$

where

$$h_q(y_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}) = \log_q\left(\frac{1}{p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y+1})}\right) + \log_q\left(\frac{1}{p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y})}\right) \tag{15}$$

$$h_q(y_{k+\delta} \mid \mathbf{y}_k^{\mu_x}) = \log_q\left(\frac{1}{p(\mathbf{y}_k^{\mu_y+1})}\right) + \log_q\left(\frac{1}{p(\mathbf{y}_k^{\mu_y})}\right)$$

In Eq. 15, the function $\log_q(x)$ is defined as [32, 37, 38]:

$$\log_q(x) = \frac{x^{1-q}-1}{1-q} \tag{16}$$

Note that using Eq. 12 and Eq. 14, we can have:

$$T_{X\to Y} = \sum_k p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}, y_{k+\delta}) t_{X\to Y}(k+\delta) \tag{17}$$

$$T_{X\to Y}^{(q)} = \sum_k p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}, y_{k+\delta}) t_{X\to Y}^{(q)}(k+\delta)$$

That is, $T_{X\to Y}$ and $T_{X\to Y}^{(q)}$ express the average information flow of the local transfer entropies (either using Shannon or Tsallis formulation) with the same probability distribution $p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}, y_{k+\delta})$.

Higher-order Tsallis local transfer entropies can also be defined:

$$t_{X\to Y}^{(q,2)}(k+\delta) = h_q^{(2)}(y_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}) \oplus_q h_q^{(2)}(y_{k+\delta} \mid \mathbf{y}_k^{\mu_x}) \tag{18}$$

where [39]

$$h_q^{(2)}(y_{k+\delta} \mid \mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y}) = \log_q p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y+1}) \oplus_q \log_q \frac{1}{p(\mathbf{x}_k^{\mu_x}, \mathbf{y}_k^{\mu_y})} \tag{19}$$

$$h_q^{(2)}(y_{k+\delta} \mid \mathbf{y}_k^{\mu_x}) = -\left(\frac{p(\mathbf{y}_k^{\mu_y+1})}{p(\mathbf{y}_k^{\mu_y})}\right)^{q-1}\left[\log_q p(\mathbf{y}_k^{\mu_y+1}) \oplus_q \log_q \frac{1}{p(\mathbf{y}_k^{\mu_y})}\right]$$

Here, the following relations are used:

$$\log_q(xy) = \log_q(x) \oplus_q \log_q(y) \tag{20}$$

$$\log_q\left(\frac{x}{y}\right) = \log_q(x) \oplus_q \log_q\left(\frac{1}{y}\right)$$

$$\log_q\left(\frac{1}{y}\right) = -y^{q-1}\log_q(y)$$

The advantages of using the (α, q)-framework are discussed in detail in Supporting Information where two coupled systems are introduced. The first describes the XOR Bernoulli-like noisy channel, which is the case of the nonlinearly coupled stochastic random processes, and the second one describes the

Gaussian-like noisy channel, which is the example linearly coupled stochastic random processes.

Following Refs. [40, 41, 13], it can be shown that for processes close to the equilibrium, both reversible or irreversible, at constant temperature (see also [42]):

$$t^{(q)}_{X\to Y}(k+\delta) = -\frac{q^{\text{ext}}_{X\to Y}}{k_B\mathcal{T}} \tag{21}$$

where $q^{\text{ext}}_{X\to Y}$ is the heat flow between the system Y and the surrounding in the context of the source X. The sign minus indicates the heat flow into the physical system Y attributed to the surrounding X and the local transfer entropy is the opposite. Eq. 21 gives the minimum amount of energy that a physical system (characterised by the dynamical component Y) must dissipate into the environment (characterised by the dynamical component X) to gain one bit of information. If $t^{(q)}_{X\to Y}(k+\delta) > 0$, and hence the physical system Y gains information, then the heat flows to the surrounding X from the system Y, $q^{\text{ext}}_{X\to Y} < 0$. In that case, a measure of the source X improves our predictions about the macroscopic transition state of Y. For thermally isolated transitions, that is $q^{\text{ext}}_{X\to Y} = 0$, then $t^{(q)}_{X\to Y}(k+\delta) = 0$, indicating that Y and X are independent, in other words, X and Y do not interact with each other.

This study concerns about the random walks of the heat flow in a graph [20], where the primary aim is the direction of the heat flow between any two vertices X and Y. For that, the following relationship between the local information flow direction and the heat flow direction is defined

$$d^{(q)}_{X\to Y}(k+\delta) = t^{(q)}_{X\to Y}(k+\delta) - t^{(q)}_{Y\to X}(k+\delta) = -\frac{\Delta q^{\text{ext}}_{X\to Y}}{k_B\mathcal{T}} \tag{22}$$

If $d^{(a,q)}_{X\to Y} < 0$, then $\Delta q^{\text{ext}}_{X\to Y} > 0$ and thus heat flow direction is from *source* X to *target* Y. In Eq. 21 and Eq. 22, $\mathcal{T}$ is the temperature (in Kelvin) and k_B is the Boltzmann's constant.

3. Molecular Dynamics Simulations of Modelled Systems

The initial protein structure for molecular dynamics simulations is the BPTI protein with PDB ID 5PTI. It contains 58 amino acids and its structure is investigated by joint X-ray and neutron experiments at a resolution of 1.00 Å [28]. The structure contains three disulphide bonds, namely Cys5-Cys55, Cys14-Cys38, and Cys30-Cys51, as depicted in Fig. 2 and visualised using Visual Molecular Dynamics (VMD) program [43].

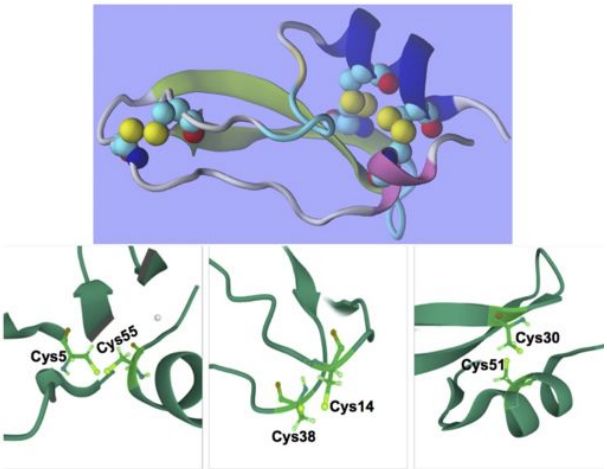

Figure 2: Three-dimensional structure and disulphide bonds of BPTI protein, visualised using VMD program [43].

Next, a partially folded, namely BPTI[14-38]Abu, mutated-type [29] with PDB ID 1YKT was considered at a resolution of 1.70 Å. BPTI[14-38]Abu has a disulphide bond Cys14-Cys38, and Cys5, Cys30, Cys51, and Cys55 are mutated to isometric α-amino-n-butyric acid. The X-ray structure misses the amino acids Glu57 and Ala58, which are modelled using a three-dimensional structure alignment with BPTI wild-type using VMD program [43] tools.

CHARMM program [44] using the CHARMM36 force field [45, 46, 47] is used for MD simulations. Figure 3 shows the topology of CHARMM36 force field for

isometric α-amino-n-butyric acid and partial charges calculated using quantum mechanics in the gas phase. Other force field parameters, such as bonds, angle bending, torsions, and van der Waals, were chosen based on the atom type as in CHARMM36 [45, 46, 47].

```
RESI ABU          0.00
GROUP
ATOM N    NH1    -0.47  !      |
ATOM HN   H       0.31  !  HN-N
ATOM CA   CT1     0.07  !      |   HB1 HG1
ATOM HA   HB1     0.09  !      |   |   |
GROUP                   !  HA-CA--CB--CG--HG3
ATOM CB   CT2    -0.18  !      |   |   |
ATOM HB1  HA2     0.09  !      |   HB2 HG2
ATOM HB2  HA2     0.09  !   O=C
GROUP                   !      |
ATOM CG   CT2    -0.18
ATOM HG1  HA2     0.06
ATOM HG2  HA2     0.06
ATOM HG3  HA2     0.06
GROUP
ATOM C    C       0.51
ATOM O    O      -0.51
```

Figure 3: The topology of CHARMM36 force field for isometric α-amino-n-butyric acid.

The hydrogen atoms and the protonation state of amino acids were determined using H++ web server [1], as in Ref. [48]. TIP3P model [49] water shell is added around the protein with a salt (NaCl) concentration of 0.154 M/l, plus the counter ions to neutralise the systems. Equilibration protocol of the solvent and hydrogen atoms (with heavy atoms of proteins fixed at their X-ray structures) included an energy minimisation (steepest descents and conjugate gradients algorithms [50] for about $5 \times 10^3 - 10 \times 10^3$ steps each), heating for about 32×10^3 $(N; V; \mathcal{T})$ MD steps (from temperature of 50 K to 800 K in steps of 50 K, constant volume V and number of atoms N) and cooling for about 40×10^3 $(N; V; \mathcal{T})$ MD steps (from 800 K to 300 K in steps of 25 K, constant V and N). Next, the restraints $(N; P; \mathcal{T})$ MD simulations at constant temperature $\mathcal{T} = 300$ K and pressure $P = 1$ atm were performed for about 500 ps for each system to equilibrate the solvent and protein by gently lowering the restraints until the density of the solvent converged to the value of 1 gram/cm^3.

[1] biophysics.cs.vt.edu

The $(N; P; \mathcal{T})$ MD simulations at constant $\mathcal{T} = 300$ K; $P = 1$ atm continued for about 2.5 ns for each system without any restraints to finalise the equilibration phase. The production run $(N; P; \mathcal{T})$ MD simulations at constant $\mathcal{T} = 300$ K; $P = 1$ atm were performed for ten ns for each system. In total, about 30 ns of simulations and around 3 months of CPU time (4 CPUs, 2.8 GHz clock rate) were estimated. A cutoff radius of $R_{\text{cut}} = 12\,\text{Å}$ was used for non-bonded long-range interactions and the SHAKE algorithm [51] to constraint the covalent bonds involving the hydrogen atoms, allowing to use a numerical integration time-step of $\Delta t = 2$ fs.

Particle-Mesh-Ewald was employed for long-range electrostatics and the potential shift for handling the short-range van der Waals interactions [42]. (Note that the van der Waals potential and the forces for this choice approach zero as the intra-distances become R_{cut}, avoiding difficult computations of switching functions and steep forces at large distances, as suggested in Ref. [44]). A fixed dielectric constant of unity was taken and simple potential cutoff approach was used for handling the short-range electrostatic interactions. The Ewald summation was performed with $\kappa = 4.0/R_{\text{cut}}$, as a trade-off between the computation cost and accuracy [42].

The cubic simulation box was used for the periodic boundary conditions, the neighbour list algorithm [52], based on the atom list, with a spherical radius of 13 Å used for keeping the non-bonded list of atoms, and the cutoff radius of the nearest neighbour was 11 Å. Here, a heuristic testing is done every time the energy and forces are calculated and a list update is done if necessary. (Note that this is both safer and more economical than frequency-updating [44]; furthermore, BYCBIM algorithm was employed to update the neighbour lists, as suggested in CHARMM manual [44]). MD simulations configurations of each system were saved every 0.2 ps and for analysis the proteins configurations were saved every 1 ps after removing the overall translational and rotational motions, based on the group of atoms chosen for further computational analysis. Wild-

type has around $N = 19562$ atoms and BPTI[14-38]Abu around $N = 18244$ atoms.

For computation of information flow, the fluctuations of coordinates of some property of interest for a group of atoms of each amino acid are stored every picoseconds as a matrix $(T \times 3n_i)$ of time series:

$$\mathbf{q}_i(k) = \langle x_1(k), y_1(k), z_1(k), \cdots, x_{n_i}(k), y_{n_i}(k), z_{n_i}(k) \rangle, \quad i = 1, \cdots, M \tag{23}$$

where n_i is the number of the group of atoms selected from the amino acid i and M is the total number of amino acids in the protein sequence. Here, k is a time index, $k = 0, 1, \cdots, T-1$ where T is the number of saved configurations. Then, we tested different properties of the amino acids, such as dipole fluctuations of the side chains, centre of the mass of amino acids, principal component analysis (PCA), and independent component analysis (ICA) to reduce the dimensionality of the vector $\mathbf{q}_i$:

$$\mathbf{X}_i(k) = \langle X_1^{(i)}(k), \cdots, X_n^{(i)}(k) \rangle, \quad i = 1, \cdots, M \tag{24}$$

where, typically, $1 \leq n \leq 3$ and $k = 0, 1, \cdots, T-1$. For the PCA and ICA computations, we used the Python software's classes *PCA* and *FastICA* of the package *sklearn.decomposition*. In our results presented in the following discussion, we used ICA projected coordinates with $n = 3$ using only the backbone atoms Cartesian coordinates of each amino acid. Then, the time delayed embedding method was used to reconstruct the dynamics of the random time processes described by the original vectors $\mathbf{q}_i(k)$, as in Ref. [20] (and the references therein), $\mu_i = (m_i, \tau_i)$. For instance, for any two amino acids i and j, we construct the vectors $\mathbf{X}_d^{\mu_{i,d}} = \{\mathbf{x}_k^{\mu_{i,d}}\}_{k=k_{0,d}}^{T-1}$, where

$$\mathbf{x}_k^{\mu_{i,d}} = \langle X_d^{(i)}(k - (m_{i,d} - 1)\tau_{i,d}), \cdots, X_d^{(i)}(k - \tau_{i,d}), X_d^{(i)}(k) \rangle,$$

and $\mathbf{Y}_d^{\mu_{j,d}} = \{\mathbf{y}_k^{\mu_{j,d}}\}_{k=k_{0,d}}^{T-1}$:

$$\mathbf{y}_k^{\mu_{j,d}} = \langle X_d^{(j)}(k - (m_{j,d} - 1)\tau_{j,d}), \cdots, X_d^{(j)}(k - \tau_{j,d}), X_d^{(j)}(k) \rangle,$$

respectively for $X_d^{(i)}$ of i and $X_d^{(j)}$ of j (for every $d = 1, 2, \cdots, n$). Here, $k_{0,d} = \max\{(m_{i,d} - 1)\tau_{i,d}, (m_{j,d} - 1)\tau_{j,d}\}$ to create vectors of the same length.

4. Results and Discussion

4.1. Structure Stability

Figure 4 shows the root mean square fluctuation (rmsf) of an amino acid in the wild-type BPTI and [14-38]Abu with Ala11Thr, Arg15Lys, Lys26Pro, and Ala27Asp mutations, based on C_α atom positions. The residues indexed between 1 and 8 show higher fluctuations in BPTI[14-38]Abu than the BPTI-WT structure, indicating a decrease in stability of the 3_{10}-helix secondary structure. Similarly, the α-helix (residues indexed between 50 and 57) becomes unstable in the structure of BPTI[14-38]Abu (as seen in Fig. 4). Besides, the root mean square fluctuation analyses indicate a decrease in stability of the residues 24-29 of the turn-like secondary structure linking the two *beta* sheets. Interestingly, the coil residues indexed 11 to 18 have fewer fluctuations in BPTI[14-38]Abu than in wild-type BPTI, indicating an increase in the stability of the first β-sheet secondary structure. The same increase in stability is observed for turn structure between residues 38 and 42 in mutated BPTI[14-38]Abu protein.

Next, the nuclear Overhauser enhancement (the so-called NOE) was computed to determine the spin-lattice dipolar relaxation contributions [53], using CHARMM program [44] implementation. Figure 5 presents the spin-spin NH−HN sequential amino acid NOEs for wild-type BPTI (bottom) and mutated BPTI[14-38]Abu (top). The most interesting is the role of residues Ala16 and Ala58. That is, in the [14-38]Abu protein NOE of Ala16 is positive, and its NOE in wild-type protein is negative; in contrast, in the [14-38]Abu protein NOE of Ala58 is negative, and it changes to a positive value in the wild-type protein. That is due to the increase of the intense of medium-range NOE of Ala16 in BPTI[14-38]Abu protein, which was confirmed by computing NH−NH non-sequential Ala16 NOEs with other amino acids; from our results, Ala16N−Cys38N proton pair was the most intense NOE of 1.077, which are two residues of the coil region linking two β-sheets. This proton pair's so-called

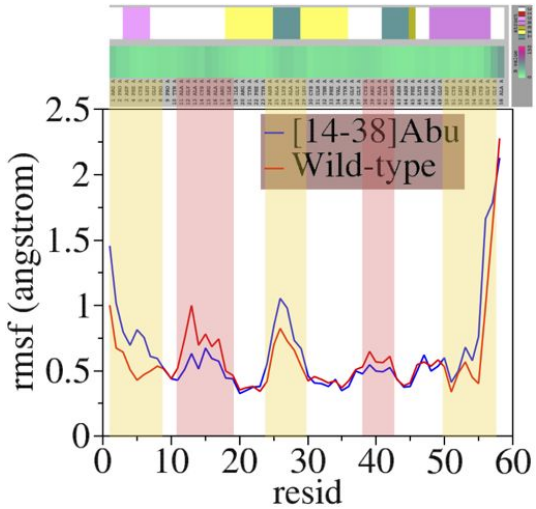

Figure 4: The rmsf of the wild-type BPTI and [14-38]Abu with Ala11Thr, Arg15Lys, Lys26Pro, and Ala27Asp mutations.

effective relaxation time was 236.89 ps in BPTI[14-38]Abu protein. Note that distance between the pair Ala16N−Cys38N from the X-ray structure is 6.83 Å [29] in mutated [14-38]Abu.

Furthermore, Table 1 shows the NMR spectra, according to formulas in reference [54]. The CHARMM program [44] was used to compute the turn secondary structure residues 25-28 using the medium-range proton pairs N−N. The equilibrium distances d are according to crystal structure experiments from references [28, 29]. Our results indicate that the NOE are more vital for the wild-type BPTI than mutated BPTI[14-38]Abu. These results support experimental findings [55] that residues 25-28 stabilise turn conformations that nucleate folding, which may prevent misfolding of β_1 and β_2 sheets.

Moreover, Table 2 presents the NMR spectra for the 3_{10}-helix secondary structure residues 3-6 using the medium-range proton pairs N−N. NHs cross peaks are sharper in the mutated [14-38]Abu protein than wild-type BPTI, indicating that these proton pairs are not in an intermediate exchange time

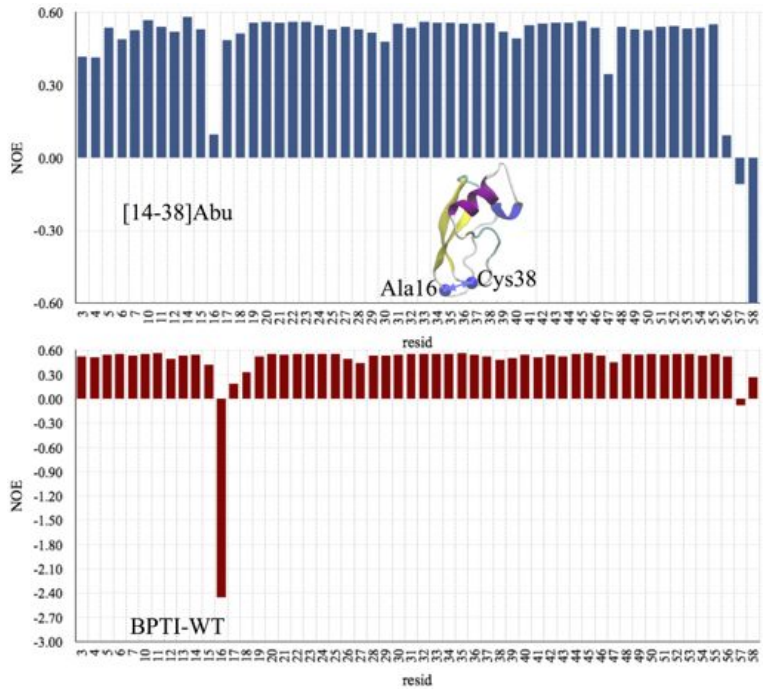

Figure 5: The NOE of the wild-type BPTI (Bottom) and [14-38]Abu with Ala11Thr, Arg15Lys, Lys26Pro, and Ala27Asp mutations (Top).

Proton pair	NOE	τ_e [ps]	TMXE [ps]	d [Å]
	BPTI-WT			
Ala25N-Lys26N	1.079	222.00	2094.0	2.72
Ala25N-Ala27N	1.078	136.00	1546.0	4.38
Ala25N-Gly28N	1.077	12.73	194.0	4.69
Lys26N-Ala27N	1.077	9.27	159.0	2.83
Lys26N-Gly28N	1.075	12.00	154.0	4.24
Ala27N-Gly28N	1.076	10.89	163.0	2.70
	BPTI[14-38]Abu			
Ala25N-Lys26N	1.075	24.49	211.0	3.04
Ala25N-Ala27N	1.071	36.29	279.0	4.77
Ala25N-Gly28N	1.075	44.59	439.0	4.55
Lys26N-Ala27N	1.074	42.33	376.0	3.01
Lys26N-Gly28N	1.077	51.49	474.0	3.99
Ala27N-Gly28N	1.076	39.12	280.0	2.62

Table 1: The NOE, effective relaxation time τ_e, and TMXE (which is the first time when the time correlation $C(t)$ is less or equal to order parameter of the spin-spin bond), according to formulas in reference [54] and computed using CHARMM program [44], for the turn secondary structure residues 25-28 using the medium-range proton pairs N−N. The equilibrium distances d are according to crystal structure experiments from references [28, 29].

regime and do not sample multiple conformations. These findings agree with NMR experimental results [55, 56].

4.2. Free Energy Change Upon Mutations

Figure 6 presents the thermodynamic cycle for calculation of binding free energy of an amino acid in a protein relative to a reference surrounding phase [57]. (Note that the same thermodynamic cycle is used in the previous studies [58, 59] using an in-house code; however, in this study, the CHARMM program [44] is

Proton pair	NOE	τ_e [ps]	TMXE [ps]	d [Å]
	BPTI-WT			
Asp3N-Phe4N	1.079	297.5	2162.0	2.83
Asp3N-Cys5N	1.078	109.0	1068.0	4.41
Asp3N-Leu6N	1.075	83.4	639.0	5.60
Phe4N-Cys5N	1.078	238.0	2246.0	2.72
Phe4N-Leu6N	1.077	132.9	1687.0	4.38
Cys5-Leu6N	1.076	55.2	949.0	2.71
	BPTI[14-38]Abu			
Asp3N-Phe4N	1.086	233.2	1032.0	2.85
Asp3N-Abu5N	1.082	182.5	996.0	4.29
Asp3N-Leu6N	1.072	83.6	304.0	5.42
Phe4N-Abu5N	1.084	351.2	2049.0	2.71
Phe4N-Leu6N	1.074	79.3	347.0	4.34
Abu5-Leu6N	1.072	71.6	349.0	2.67

Table 2: The NOE, effective relaxation time τ_e, and TMXE, according to formulas in reference [54] and computed using CHARMM program [44], for the 3_{10}-helix secondary structure residues 3-6 using the medium-range proton pairs N−N. The equilibrium distances d are according to crystal structure experiments from references [28, 29].

used). The differences in the solvation free energy ($\Delta G_{(k)}^{(\text{solvation})}$, $\Delta G_{(-k)}^{(\text{solvation})}$, and $\Delta G^{(\text{solvation})}$) in reference surrounding phase (often gas phase) and in a solvent (such as salted water) are computed using Poisson-Boltzmann (PB) method [42] implemented in CHARMM [44]. Here, $\Delta G_k^{(\text{b,solv})}$ and $\Delta G_k^{(\text{b,ref})}$ denotes the binding free energies of the amino acid k in the solvent and gas phase, respectively. $\Delta G_k^{(\text{b,solv})}$ is computed as:

$$\Delta G_k^{(\text{b,solv})} = \Delta G_k^{(\text{b,ref})} + \Delta G_k^{(\text{solvation})} \tag{25}$$

where

$$\Delta G_k^{(\text{b,ref})} = H^{(\text{MM})} - H_{(-k)}^{(\text{MM})} - \mathcal{T}\Delta S_k \tag{26}$$

$$\Delta G_k^{(\text{solvation})} = \Delta G^{(\text{solvation})} - \Delta G_{(-k)}^{(\text{solvation})}$$

Here, $H^{(\text{MM})}$ and $H_{(-k)}^{(\text{MM})}$ represent the potential interaction energy of the entire system and subsystem without amino acid k, based on the CHARMM36 molecular force field. Furthermore, ΔS_k is the change on the entropy with $\mathcal{T}$ being the temperature (typically, $\mathcal{T} = 300$ K, the same as in MD simulations), which is decomposed into three terms, namely rotational, translational and configurational entropies, implemented in CHARMM [44]. Note that, here, Andricioaei & Karplus formula is used to compute configurational entropy [60, 61].

Moreover, $\Delta G_k^{(\text{solvation})}$ is decomposed into the polar (or electrostatic) and non-polar (or solvent-accessible-surface area, SASA) terms as:

$$\Delta G_k^{(\text{solvation})} = \Delta G_{\text{polar},k}^{(\text{solvation})} + \Delta G_{\text{SASA},k}^{(\text{solvation})} \tag{27}$$

In Eq. 27, $\Delta G_{\text{polar},k}^{(\text{solvation})}$ is calculated using PB method and $\Delta G_{\text{SASA},k}^{(\text{solvation})}$ is parametrised as

$$\Delta G_{\text{SASA},k}^{(\text{solvation})} = \gamma A_k \tag{28}$$

where A_k is the value of SASA for the amino acid k present in the protein and $\gamma = 0.006$ kcal $\times$ mol^{-1} $\times$ Å^{-2}. The coefficient γ and parametrisation of the surface effects are designed such to include short-range effects, dominated by the first solvation shell [62, 63]. There is a wide range of the values of γ because of different models used to represent the molecular surface [62]. However, small values of γ indicate that molecular surface contribution has only a small impact on the free energy changes due to large-scale conformation changes. Note that all the computed quantities are time averaged ones over different configurations obtained from MD simulations; typically, 2000 configurations were used. The statistical analyses use an 95 % confidence interval for the mean [64]. All the CHARMM's scripts for these statistical analyses and computations are specifically created in this study with the help of CHARMM [44] manual and Forum

website [2] over several years.

The computations included each system (such as BPTI-WT and BPTI[14-38]Abu proteins). The differences between the free energy changes due to mutations are calculated as:

$$\Delta\Delta G^{(\mathrm{b,solv})}_{k\to k'} = \Delta G^{(\mathrm{b,solv})}_{k'} - \Delta G^{(\mathrm{b,solv})}_{k} \tag{29}$$

where k is the amino acid in BPTI-WT and k' is the amino acid with the same ID in BPTI[14-38]Abu.

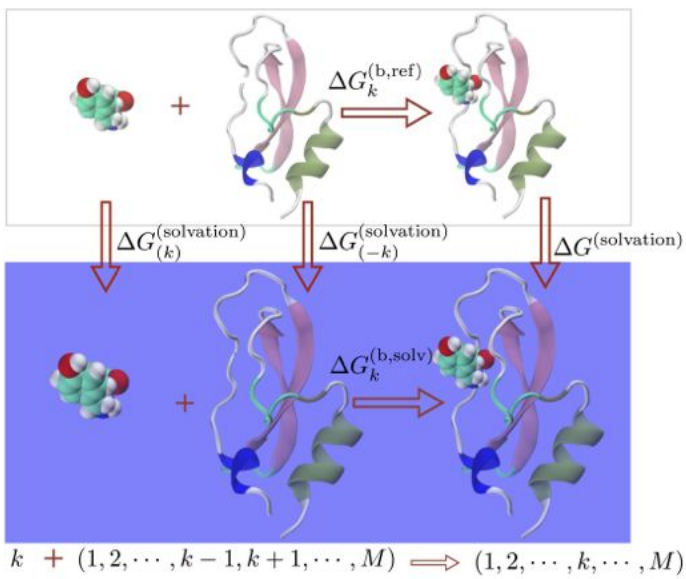

Figure 6: Thermodynamic cycle for calculation of binding free energy of an amino acid in a protein. The differences in the solvation free energy, $\Delta G^{(\mathrm{solvation})}_{(k)}$, $\Delta G^{(\mathrm{solvation})}_{(-k)}$, and $\Delta G^{(\mathrm{solvation})}$ in reference surrounding phase (often gas phase) and in solvent (such as water and salt).

Figure 8 shows the results for $\Delta\Delta G^{(\mathrm{b,solv})}_{k\to k'}$. In the computations of the electrostatic Poisson-Boltzmann terms, the dielectric constants of the protein and water are, respectively, $\epsilon_p = 2$ and $\epsilon_w = 80$. The solvent salt concentration

[2] www.charmm.org

was 0.154 M/l (the same as in atomistic MD simulations). From our calculations, mutation of Cys5 and Cys55 to Abu resulted in favour of the stability of the structure, respectively, $\Delta\Delta G^{(\mathrm{b,solv})}_{\mathrm{Cys5\to Abu5}} = -3.185 \pm 0.484$ kcal/mol and $\Delta\Delta G^{(\mathrm{b,solv})}_{\mathrm{Cys55\to Abu55}} = -3.701 \pm 0.470$ kcal/mol. On the other hand, the mutations of Cys30 and Cys51 to Abu favoured much less the stability of the BPTI[14-38]Abu; in particular, $\Delta\Delta G^{(\mathrm{b,solv})}_{\mathrm{Cys30\to Abu30}} = -0.724 \pm 0.463$ kcal/mol and $\Delta\Delta G^{(\mathrm{b,solv})}_{\mathrm{Cys51\to Abu51}} = -0.447 \pm 0.551$ kcal/mol. Interestingly, the most favoured mutation in enhancing the stability of BPTI[14-38]Abu structure is Ala27→Asp27 with a value of $\Delta\Delta G^{(\mathrm{b,solv})}_{\mathrm{Ala27\to Asp27}} = -13.784 \pm 0.441$ kcal/mol. In contrast, the mutation of Lys26 to Pro26 disfavoured the stability with a value of $\Delta\Delta G^{(\mathrm{b,solv})}_{\mathrm{Lys26\to Pro26}} = 2.856 \pm 0.606$ kcal/mol; similarly the mutation of Arg15 to Lys15 with a value of $\Delta\Delta G^{(\mathrm{b,solv})}_{\mathrm{Arg15\to Lys15}} = 2.377 \pm 0.939$ kcal/mol.

Furthermore, the mutation of Ala11 to Thr11 favoured the stability of BPTI[14-38]Abu structure with a change on the free energy of -3.708 ± 0.400 kcal/mol. Interestingly, other amino acids, which are not involved in the mutation, contribute to the stability of the protein structure. This indicates that the interaction network topology between the amino acids in the protein sequence has possibly changed upon the mutations. Therefore, the following graph information flow network analyses could potentially enable identification of the changes in the amino acids interaction networks of both proteins. The aim is to reveal the encoded information of the sequence of amino acids into a protein structure using the newly developed (α, q) information-theoretic measures. In particular, the primary interest is to investigate statistical efficiency of the Sharma-Mittal transfer entropy, based on the (α, q)-framework.

4.3. The (a, q)-Framework Directional Information Flow

In this study, the information flow direction is calculated from the Sharma-Mittal symbolic encoding transfer entropies, based on the (a, q)-framework, as:

$$D^{(a,q)}_{i\to j} = T^{(a,q)}_{i\to j} - T^{(a,q)}_{j\to i}, \quad i, j = 1, 2, \cdots, M \tag{30}$$

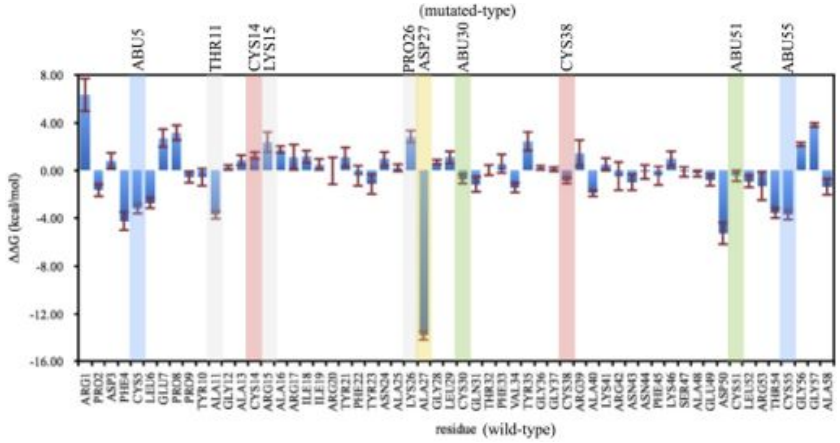

Figure 7: The free energy changes for each amino acid upon mutation from protein BPTI-WT to BPTI[14-38]Abu.

where M is the number of amino acids in the structure. The choice was $\alpha = 2.0$ and $q = 0.85$, based on the analysis presented in Supporting Information material associated with study. Here, if $D_{i\to j}^{(a,q)} > 0$, then information flows from source i to target j, or i influences j. Figure 8 presents heat map plots of $D_{i\to j}^{(a,q)}$ (in nats) for BPTI-WT and BPTI[14-38]Abu, where a red colour indicates a strong *source* of information and blue one indicates a *target* (or *sink*) amino acid of information flow. Our results clearly identify the source and target residues and the change in the amino acid's propensity to be a *source* or *target* (*sink*) of information flow for BPTI-WT and BPTI[14-38]Abu.

Figure 9 shows the differences in information flow for each amino acid in BPTI wild-type and BPTI[14-38]Abu structures (Fig. 9(a)) calculated as:

$$\Delta D_{i\to j}^{(a,q)} = D_{i\to j}^{(a,q),\mathrm{wt}} - D_{i\to j}^{(a,q),\mathrm{mt}}, \quad i,j = 1,2,\cdots,M \tag{31}$$

Here, if $\Delta D_{i\to j}^{(a,q)} > 0$, then amino acid i has a greater propensity to be a source of information than j in wild-type structure relative mutated protein, which is indicated in Fig. 9(a) by a red colour; otherwise, if $\Delta D_{i\to j}^{(a,q)} < 0$, then i has a greater propensity to be a target of information than j in wild-type protein

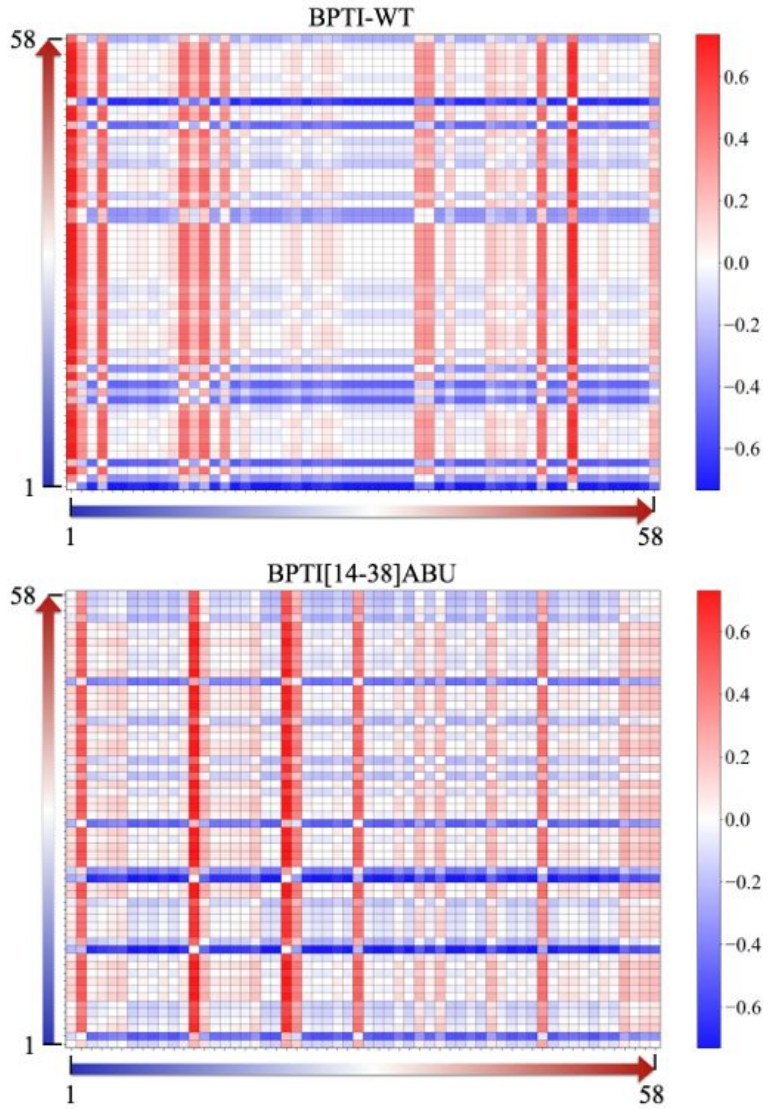

Figure 8: The directional information flow (in nats) for BPTI wild-type (Top) and BPTI[14-38]Abu (Bottom).

relative to mutated one, indicated in Fig. 9(a) by a blue colour. In Fig. 9, we have depicted the amino acids with the greatest positive impact as a *source* of information flow, such as Ala13, Phe22, Tyr23, Leu29, Gly37, Thr54, and Cys55(wild-type)/Abu55(mutated-type). Furthermore, the amino acids with the greatest positive impact as a *sink* of information flow include: Arg1, Phe4, Ala11(wild-type)/Thr11(mutated-type), Gly12, Cys14, Ala16, Gln31, Gly36, Cys38, and Asp50.

We also calculated the mean-square-deviation (msd) of $D_{i\to j}^{(a,q)}$ between BPTI-WT and BPTI[14-38]Abu as:

$$\mathrm{msd}(i) = \frac{1}{M}\sum_{j=1}^{M}\left(D_{i\to j}^{(a,q),\mathrm{wt}} - D_{i\to j}^{(a,q),\mathrm{mt}}\right)^2 \tag{32}$$

The results are presented graphically in Fig. 9(b), where one can clearly identify the amino acids with the largest deviation in $D_{i\to j}^{(a,q)}$ from wild- to mutated-type protein. Interestingly, there is some relationship between the changes on $D_{i\to j}^{(a,q)}$ and the changes on the free energies (magnitude); for example, large values of msd for residues 1, 4, 11, 50, and 55 correspond to significant changes on the free energies. That may indicate that the information-theoretic measure $D_{i\to j}^{(a,q)}$ (based on the fluctuations of the encoded information) captures the nature of interactions within a protein structure, which is highly non-linear based on the physics-nature of molecular force fields [3]. This is possibly related to the fact that (α, q)-framework maximises the average information flow between the Markovian time processes, as was analysed in the Supporting Information for more intuitive non-linearly coupled random Markovian processes (such as the input-output of the XOR Bernoulli-like noisy channel).

4.4. Amino Acids Ranking

Next, we considered graph-theoretic measures (such as PageRank, Authority, and Hub) [65, 66, 67, 68, 69, 70] to rank the impact of the amino acids in the protein's interaction network; for that, first, a DiGraph is constructed

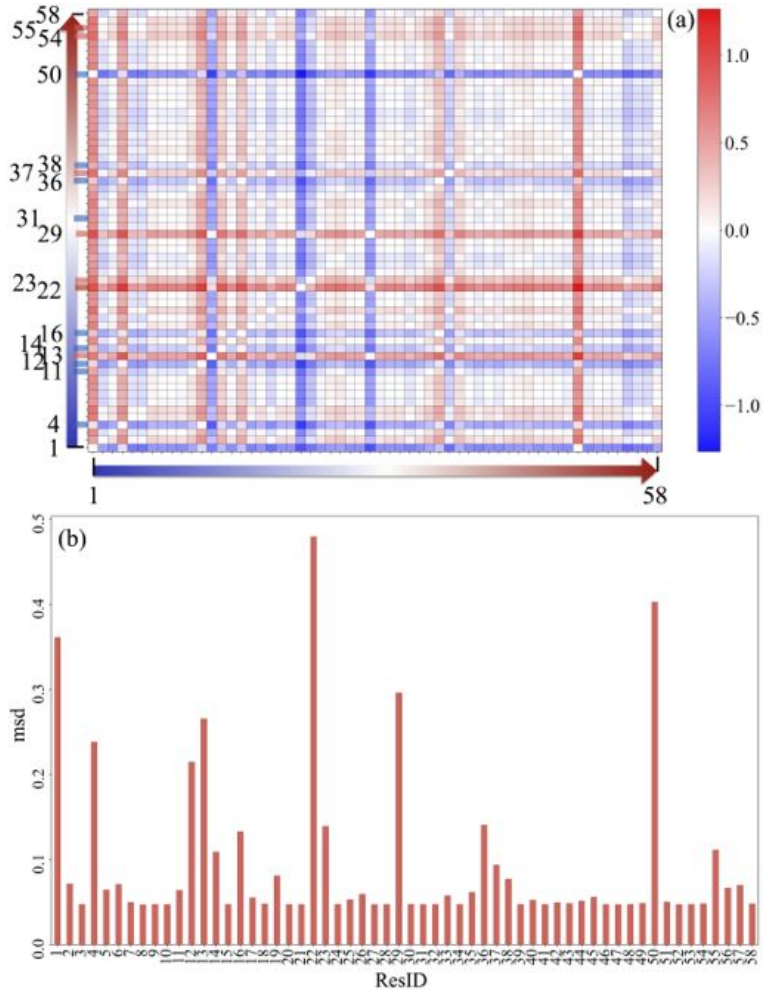

Figure 9: (a) The directional information flow difference between BPTI wild-type and BPTI[14-38]Abu; (b) The msd of the information flow difference between BPTI wild-type and BPTI[14-38]Abu.

using directional information flow matrix $D_{i\to j}^{(a,q)}$. For the PageRank, the algorithm introduced in Ref. [20] is used. For the Authority and Hub ranks, a new algorithm is developed based on the so-called Google matrix, computed as in Ref. [20]. The algorithm uses the Google matrix L as an input, then, iteratively, the Authority rank (a_i, for $i = 1, 2, \cdots, n$) and Hub rank (h_i, for $i = 1, 2, \cdots, n$) scores are computed for each vertex i of the DiGraph G [68]:

$$\begin{aligned} \mathbf{h}_{t+1} &= \mathbf{L}^T \mathbf{a}_t \\ \mathbf{a}_{t+1} &= \mathbf{L}\mathbf{h}_t, \quad t = 0, 1, \cdots, N_{\text{iter}} \end{aligned} \tag{33}$$

where $()^T$ stands for the transpose, t is an iteration index, and N_{iter} is the number of iterations (maximum 100 considered in this study). At each iteration, the vectors $\mathbf{a} = (a_1, a_2, \cdots, a_n)^T$ and $\mathbf{h} = (h_1, h_2, \cdots, h_n)^T$ are normalised to one. Here, the Authority score indicates the rank of an amino acid as a *source* of information flow through the graph and the Hub scoring ranks the vertices (and thus the amino acids) based on their influence as *mediators* or *gateways* of the information flow along a path (identified in this study as the *shortest* and the *best* paths, as described below) from a *source* to the corresponding *target*.

Our results of the PageRank and Hub ranks for each amino acid of BPTI wild-type and BPTI[14-38]Abu are shown in Fig. 10. The PageRank and Hub scores may show a helpful trend with the changes in the free energy from the wild- to the mutated type. In particular, an increase in the PageRank's rank of an amino acid, relative to others, from BPTI-WT to BPTI[14-38]Abu results in a favourable free energy change of that amino acid in mutated structure. For instance, mutation of Ala27 to Asp27 increases the rank of Asp27 (and so its influence) in BPTI[14-38]Abu, and a favourable free energy change is also observed. Similar results are obtained for amino acids 4, 11, 14, 15, 26, 38, and 50, as indicated in Fig. 10. Interestingly, the same trend is observed when the Hub rank increases; in particular, the increase of the Hub rank scores for the amino acids with ID 4, 5, 6, 30, 54, and 55, relative to others in the same protein structure from the wild- to mutated-type protein, correlates well with the

corresponding changes on the free energy. That may have a significant impact on predicting the amino acids that, if mutated, could result in a favourable free energy change due to their cooperative contribution within the structure as a mediator of the directional information flow in the protein interaction network of amino acids.

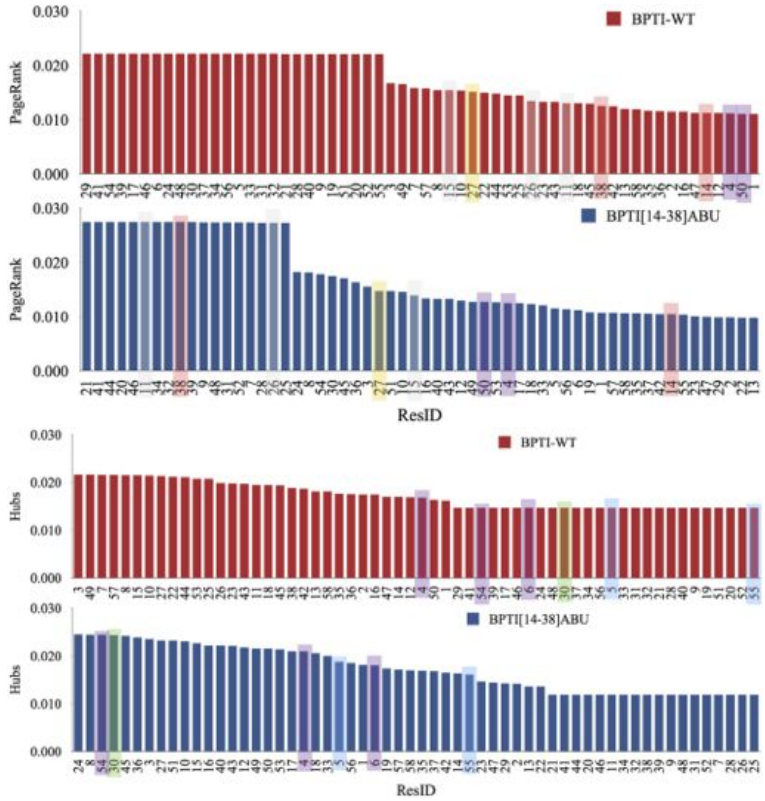

Figure 10: The PageRank and Hub ranking for each amino acid of BPTI wild-type and BPTI[14-38]Abu.

Authority rank of amino acids showed strong correlation with PageRank, as presented in Fig. 11. Therefore, our study indicates that these two DiGraph interaction network measures characterise similar topological properties and their can be used interchangeably.

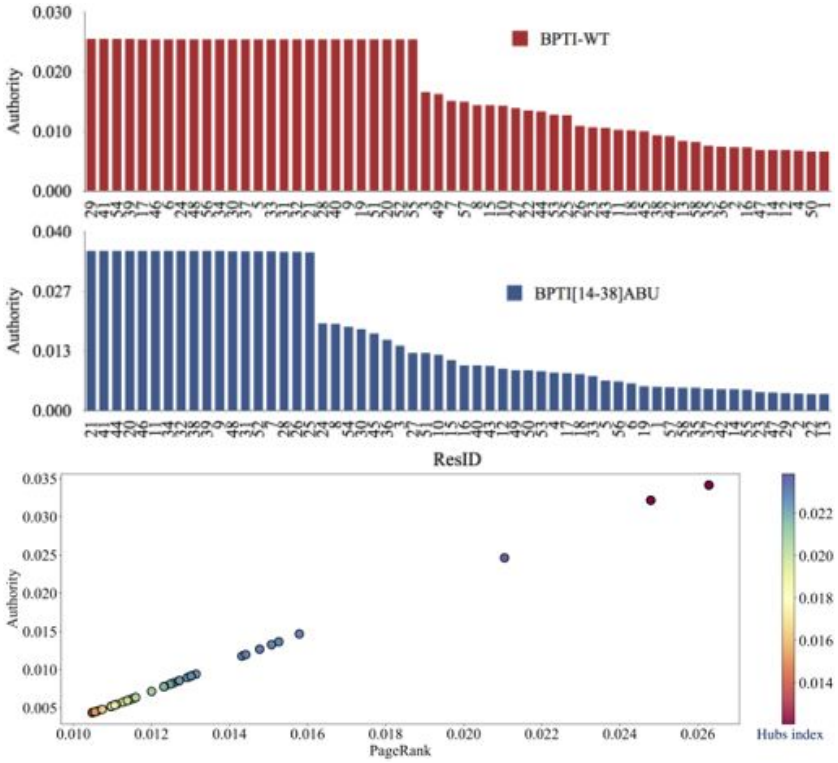

Figure 11: The authority ranking for each amino acid of BPTI wild-type and BPTI[14-38]Abu (two plots in the top) and the correlation between the PageRank and Authority indices (the plot in the bottom).

4.5. Amino Acids Information Flow Pathways

The above graph-theoretic measures can also identify the shortest and best paths between a *source* and *target* amino acid.s Here, the algorithm, developed in Ref. [34], is used to determine the shortest and the best paths of information flow in an amino acids' interaction DiGraph network topology. In Fig. 12, we present the shortest/best paths of amino acids between Cys14 and Cys38 for BPTI wild-type and BPTI[14-38]Abu. The algorithm given in Ref. [71] was used, as implemented in Ref. [34]. The paths are identified by entering the first and the last amino acids, representing, respectively, the so-called above the *target* and *source* of information flow; practically, this is determined by comparing the PageRank ranks, PageRank(first) $<$ PageRank(last), or information flows from the last to the first. Figure 11 shows different possible paths between a target and source of the amino acids for BPTI-WT and BPTI[14-38]Abu. All the intermediate amino acids act as hubs (mediators or gateways of the directional information flow from a source to the corresponding target amino acid). It is worth mentioning that all paths for wild- and mutated-type proteins pass through the same number of hubs; however, the mediator role of the Hubs is undertaken by different amino acids. Again, this indicates the changes occurring in the interaction network DiGraph and its directional information flow topology due to the mutation of amino acids.

Figure 13 presents a long path and the corresponding short path of amino acids between Cys14 and Cys38 for BPTI wild-type (see Top plot) and the shortest path between Abu55 and Abu5 for BPTI[14-38]Abu. This kind of examination can be used to identify the most influential Hubs from the source to target in an interaction network DiGraph. Pro2 is potentially the most influential gateway of the information flow between Cys14 and Cys38.

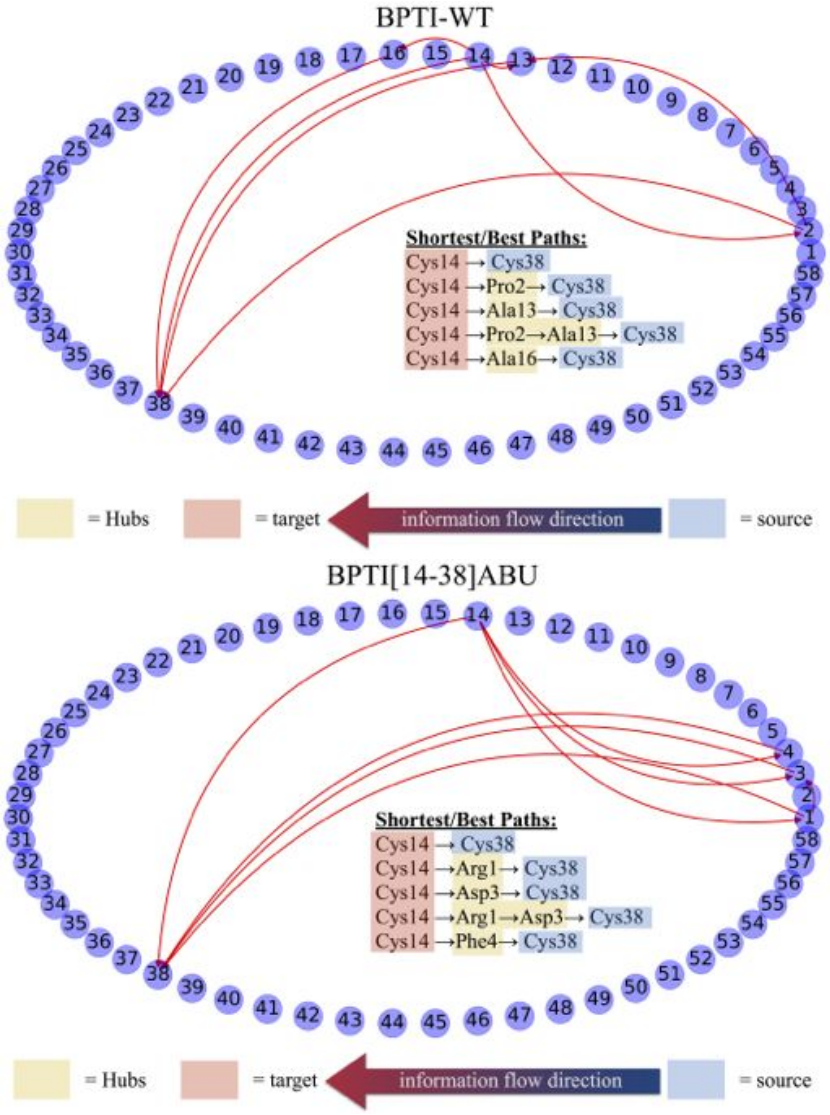

Figure 12: The shortest/best paths of amino acids between residues Cys14 and Cys38 for BPTI wild-type and BPTI[14-38]Abu. The algorithm given in Ref. [71] and implemented in [34] was used.

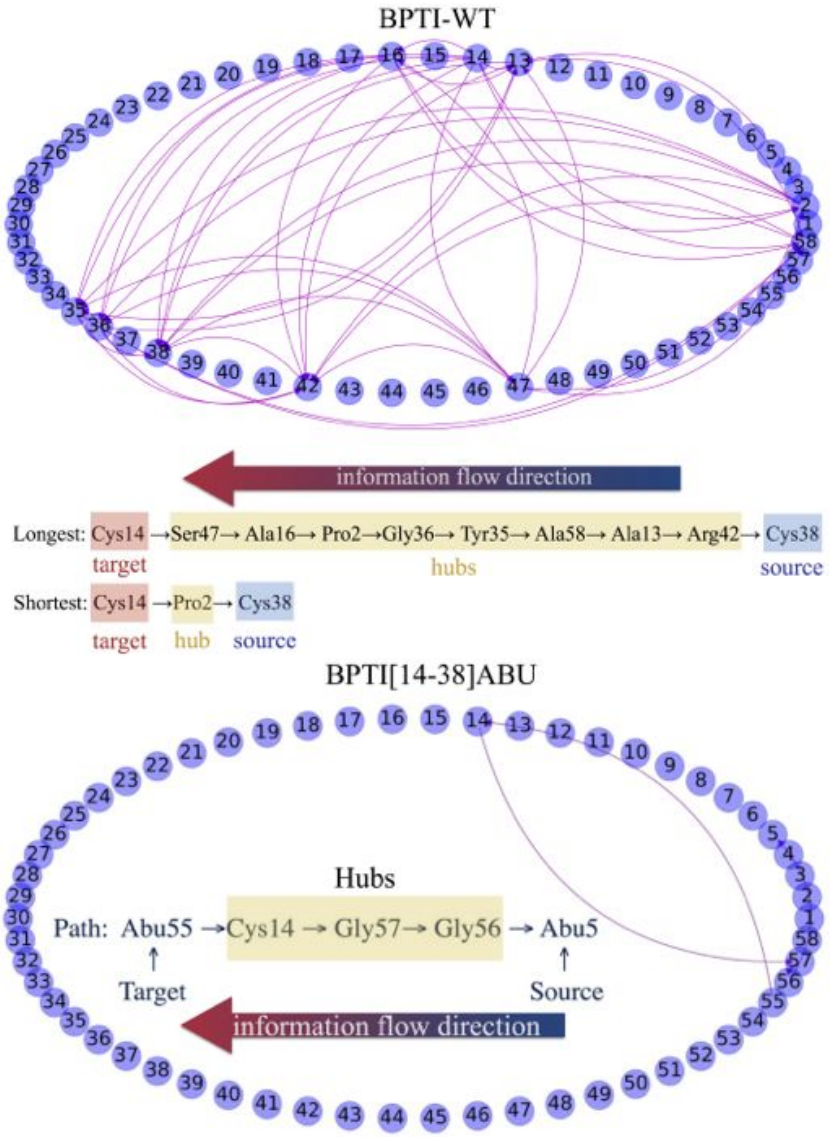

Figure 13: (Top) A long and short path of amino acids between Cys14 and Cys38 for BPTI wild-type; (Bottom) A short path between Abu55 and Abu5 for BPTI[14-38]Abu.

4.6. Machine Learning of Amino Acids Ranks

4.6.1. Heat Flow Kernel DiGraph Embedding

Consider a directed graph $G(V, E, \mathbf{A})$ with M vertices, $V = \{v_1, v_2, \cdots, v_M\}$, and m edges, $E = \{e_1, e_2, \cdots, e_n\}$ where $e_i = (u, v)$ is a directed edge from u to v, characterised by the weight W_{uv} and an adjacency matrix $\mathbf{A}$:

$$A_{uv} = \begin{cases} W_{uv} & \text{if } (u, v) \in E \\ 0 & \text{otherwise} \end{cases} \tag{34}$$

Next, two diagonal matrices can be determined to characterise the in- and out-degrees of vertices, denoted $\mathbf{D}^{(\text{in})}$ and $\mathbf{D}^{(\text{out})}$, respectively, as:

$$\mathbf{D}^{(\text{in})} = diag\left(d_1^{(\text{in})}, \cdots, d_u^{(\text{in})}, \cdots, d_M^{(\text{in})}\right), \quad d_u^{(\text{in})} = \sum_{v=1}^{M} A_{vu} \tag{35}$$

$$\mathbf{D}^{(\text{out})} = diag\left(d_1^{(\text{out})}, \cdots, d_u^{(\text{out})}, \cdots, d_M^{(\text{out})}\right)^T, \quad d_u^{(\text{out})} = \sum_{v=1}^{M} A_{uv} \tag{36}$$

Then, the transition probability matrix $\tilde{\mathbf{P}}$ of rank $M \times M$ of a random walk on a DiGraph is given as:

$$\tilde{\mathbf{P}} = \left[\mathbf{D}^{(\text{out})}\right]^{-1} \mathbf{A} \tag{37}$$

For a strongly connected digraph the random walk characterised by $\tilde{\mathbf{P}}$ is an irreducible and aperiodic Markovian transition process, which is satisfied if there are no vertices with $d_u^{(\text{out})} = 0$ [20]. The following transition probability matrix is suggested to guarantee the irreducibility and aperiodicity

$$\mathbf{P} = (1 - \alpha)\left[\mathbf{D}^{(\text{out})}\right]^{-1} \mathbf{A} + \alpha \mathbf{J} \tag{38}$$

where $\mathbf{J}$ is a jumping matrix given in terms of the unitary diagonal matrix $\mathbf{1}_M$ of rank $M \times M$ as:

$$\mathbf{J} = \frac{1}{M} \mathbf{1}_M \tag{39}$$

Here, α is a number between 0.1 and 0.2 [20] (and the references therein). Then, the normalised Laplacian matrix is constructed as [26]:

$$\begin{aligned}\mathbf{L} &= \left[\mathbf{D}^{(\text{out})}\right]^{-1/2}\left[\mathbf{D}^{(\text{out})} - \mathbf{D}^{(\text{out})}\mathbf{P}\right]\left[\mathbf{D}^{(\text{out})}\right]^{-1/2} \\ &= \left[\mathbf{D}^{(\text{out})}\right]^{+1/2}\left[\mathbf{1}_M - \mathbf{P}\right]\left[\mathbf{D}^{(\text{out})}\right]^{-1/2}\end{aligned} \tag{40}$$

Spectral decomposition of $\mathbf{L}$ matrix is given as:

$$\mathbf{L} = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^T \tag{41}$$

where $\mathbf{E}$ is the matrix of ordered eigenvectors, $\mathbf{e}_u = (E_{1u}, \cdots, E_{Mu})^T$ as columns and $\mathbf{\Lambda}$ is a diagonal matrix of ordered eigenvalues ($\lambda_1 < \lambda_2 < \cdots < \lambda_M$), which are all non-negative.

The vertices of the DiGraph G are mapped into a vector space using the heat kernel such that for each vertex, we determine a so-called embedded vector $\mathbf{z}_u^{(t)}$ of M dimensions at any time t (for $u = 1, 2, \cdots, M$) [26]:

$$\mathbf{z}_u^{(t)} = \left(\exp\left(-\frac{1}{2}\lambda_1 t\right)E_{1u}, \exp\left(-\frac{1}{2}\lambda_2 t\right)E_{2u}, \cdots, \exp\left(-\frac{1}{2}\lambda_M t\right)E_{Mu}\right)^T \tag{42}$$

where t is the random walk diffusive time of heat flow on the directed graph. That represents a kernel mapping

$$\mathcal{F} : V \longrightarrow R^M, \tag{43}$$

which embeds each vertex of the DiGraph into a vector space R^M. Thus, the Euclidean distance between any two vertices u and v is defined as:

$$\ell(u, v) = \sqrt{(\mathbf{z}_u^{(t)} - \mathbf{z}_v^{(t)})^T(\mathbf{z}_u^{(t)} - \mathbf{z}_v^{(t)})} \tag{44}$$

4.6.2. Deep Graph Convolution Neural Network Model

First, we describe the graph representation of the dataset created by computation of the heat flow $q_{u\to v}^{\text{ext}}$ between two amino acids u and v, representing the

vertices of a DiGraph. That is done at regular time intervals of MD simulations; therefore, a set of such graph representations will be stored in the dataset with $t = 0, 1, \cdots, T-1$ entries where T is the number of snapshots saved from a typical MD simulation.

Figure 14(a) illustrates a graph representation (adapted from Ref. [25]) of the heat flow at some time frame t, $G_t(V, E, \mathbf{A})$, in a protein where vertices represent amino acids and the arrow lines indicate the direction of heat flow; that is, if $q^{\text{ext}}_{u\to v} > 0$, an edge is added from u to v with weight $W_{uv} = q^{\text{ext}}_{u\to v}$. This constructs an adjacency matrix $\mathbf{A}$, as illustrated in Fig. 14(b). Each vertex u is associated with an embedding vector $\mathbf{z}_u$, constructing the embedding matrix of rank $M \times M$ (see Fig. 14(c)), which is computed from spectral decomposition of the Laplacian $\mathbf{L}$ matrix using Eq. 42 and $\mathbf{L}$ is computed using Eq. 40. Besides, one could assign an embedding matrix to the edge attributes; here, the probability transition matrix $\mathbf{P}$, given by Eq. 38, is used as the edge embedding matrix, and hence $n = 1$.

A graph neural network (GNN) is a computation model that uses the graph representation described above as an input (that is, the vertex embedding vectors $\mathbf{z}_u$, $u = 1, 2, \cdots, M$, adjacency matrix $\mathbf{A}$, and edge embedding matrix $\mathbf{P}$) and transfers it into a set of L layers. The vertex embedding vectors $\mathbf{z}_u$, at every layer, update to construct the so-called intermediate *hidden* representations $\mathbf{h}_{u,l}$ with the output vertex embedding vector $\mathbf{h}_{u,L}$. The GNN uses matrix representation; $\mathbf{Z}$ is the matrix of vertex embeddings with the columns the vertex embeddings $\mathbf{z}_u$. Similarly, the layer embeddings $\mathbf{H}_l$ and the output matrix $\mathbf{Y}$ are defined. The input layer of the GNN contains the information about the embeddings of each vertex u in $\mathbf{Z}$ and the last layer (layer L) contains the information about the vertices and their context within the graph in $\mathbf{H}_L$.

In this study, we assign to each vertex u of the graph an output vector $\mathbf{y}_u$, representing the set of measurements to be predicted using the so-called *regression* at a vertex-level task. In particular, we used the amino acids ranks

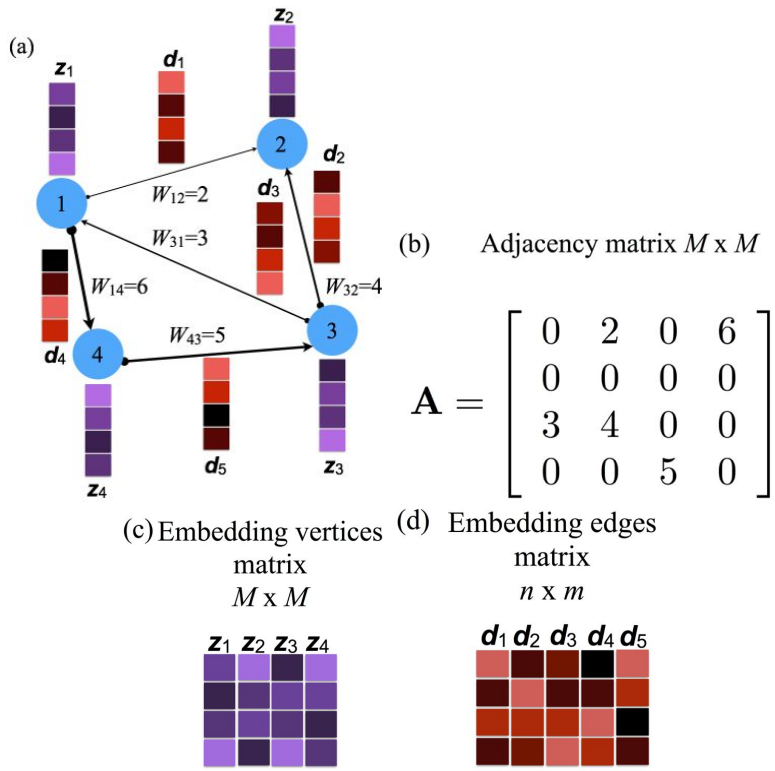

Figure 14: a) Illustration of a graph representation indicating the vertices set, edges set, edges weights, edges embeddings, vertex embeddings; (b) Adjacency matrix $M \times M$; (c) Embedded vertices matrix $M \times M$; and (d) Embedded edges matrix $n \times m$. It is slightly adapted from Ref. [25].

(such as PageRank, Authority, and Hub ranks):

$$\mathbf{y}_u^{(t)} = (\text{PageRank}(u),\ \text{Authority}(u),\ \text{Hub}(u))^T \tag{45}$$

That is done using the graph structure and vertex embeddings. Besides, the loss functions are independently defined at each vertex u:

$$\mathbf{y}_u^{(\text{pred})} = \sigma\left(\boldsymbol{\gamma}_L^{(1)} + \boldsymbol{\gamma}_L^{(2)}\mathbf{h}_{u,L}\right) \tag{46}$$

where σ is a sigmoid or softmax function and the column vector $\boldsymbol{\gamma}_L^{(1)}$ and a matrix $\boldsymbol{\gamma}_L^{(2)}$ are learned parameters. (Note that a final linear transformation is sometimes needed to map vertex embeddings to the expected output feature dimensionality). The loss function is the mean square error (mse). One can also use the same network to predict the edge attribute embeddings (such as the instant directional information flow $d_{u\to v}^{(\alpha,q)}$) between any two vertices u and v, such as the transition probability matrix elements [25]:

$$P^{(\text{pred})}(v|u) = \sigma\left((\mathbf{h}_{u,L})^T\mathbf{h}_{v,L}\right) \tag{47}$$

This study uses the graph convolutional network (GCN) among the other GNNs as a computational model [25]. The convolutional neural networks have been subject to other studies [72]. In a GCN model, the vertices are updated by aggregation of the information from the neighbouring vertices by considering a relational inductive bias. Here, the spacial-based GCN model is implemented that applies convolution spacial domain based on the initial graph structure, mathematically modelled in Ref. [25]. A nonlinear activation function f (such as a sigmoid or softmax) applies at each layer for each vertex u with parameters $(\mathbf{b}_l, \boldsymbol{\Gamma}_l)$ (for $l = 0, \cdots, L$), where the inputs are the vertex embeddings and adjacency matrix. The activation function applies to every vertex embedding as [25]:

$$\begin{aligned} \mathbf{h}_{u,1} &= f\left(\mathbf{b}_0 + \boldsymbol{\Gamma}_0^T\left[\mathbf{z}_u + \mathbf{a}(u,1)\right]\right) \\ \mathbf{h}_{u,l} &= f\left(\mathbf{b}_{l-1} + \boldsymbol{\Gamma}_{l-1}^T\left[\mathbf{h}_{u,l-1} + \mathbf{a}(u,l-1)\right]\right), \quad l = 2, \cdots, L \end{aligned} \tag{48}$$

where $\mathbf{a}(\cdots)$ is the aggregation function:

$$\mathbf{a}(u,l) = \sum_{v \in N(u)} \mathbf{h}_{v,l} \tag{49}$$

where $N(u)$ is the list of the neighbouring vertices of u. The outputs are again predicted as follows:

$$\mathbf{y}_u^{(\mathrm{pred})} = \sigma\left(\boldsymbol{\gamma}_L^{(1)} + \boldsymbol{\gamma}_L^{(2)} \mathbf{h}_{u,L}\right) \tag{50}$$

In our training, we considered a dataset of T graphs $G_t(V, E, \mathbf{A}, \mathbf{P})$ and the measurements $\mathbf{Y}^{(t)}$ $(t = 0, 1, \cdots, T-1)$. The dataset is randomly split into training dataset (80 %) T_{train}, validation dataset (10 %) T_{valid}, and test dataset (10 %) T_{test}. The test dataset is such that it does not share any information with training or validation datasets. In this way, the set of training parameters consists of $\{\mathbf{b}_l, \boldsymbol{\Gamma}_l\}$ for $l = 0, 1, \cdots, L$. We used the so-called *Adam* algorithm and the MSE loss function as an optimiser. The mini-batches are chosen as a random subset of graph vertices at each training iteration. Other aggregation functions are also chosen for testing, such as the mean or maximum pooling aggregation functions:

$$\mathbf{a}(u,l) = \frac{1}{\mid N(u) \mid} \sum_{v \in N(u)} \mathbf{h}_{v,l} \tag{51}$$

$$\mathbf{a}(u,l) = \max_{v \in N(u)} \mathbf{h}_{v,l} \tag{52}$$

A jump knowledge connection algorithm [73] was used, according to which the model will additionally apply a final linear transformation to transform vertex embeddings to the expected output feature dimensionality. The long short-term memory (lstm) algorithm [74] was used. Furthermore, a DeeperGCNLayer model [75] was implemented, including the pre-activation residual connection, the residual connection, the dense connection, and no connections.

4.6.3. Test Case

The first test system is an intuitive system of five vertices, with a local transfer entropy given as:

$$\left[t_{i\to j}^{(0)}\right] = \begin{bmatrix} 0.0 & 6.1 & 7.5 & 6.5 & 6.7 \\ -0.2 & 0.0 & -0.2 & -0.2 & 0.2 \\ 6.0 & 7.1 & 0.0 & 4.5 & 7.0 \\ -0.1 & -0.2 & -0.2 & 0.0 & 0.2 \\ 2.2 & 0.8 & 0.2 & 0.2 & 0.0 \end{bmatrix} \tag{53}$$

At every time iteration, a Gaussian-like noise $\mathcal{N}(0,1)$ (mean value zero and standard deviation one) is added to every element of the matrix as:

$$t_{i\to j}^{(t)} = t_{i\to j}^{(0)} + 0.01 \cdot \mathcal{N}(0,1) \tag{54}$$

It is easy to identify which vertex 1 has the highest rank and which vertex 3 has the second highest. The length of the dataset is $T = 2000$ frames, and the lengths of the training, validation, and test datasets are 80 %, 10 %, and 10 %, respectively. The batch sizes are $(100, 1, 1)$ and several epochs of 250 iterations on the training dataset. In this case, six layers with 32 channels, each layer is used as a setup model. The *adam* optimiser is chosen, the activation function for the hidden layers is the so-called *relu*, and the sigmoid function for the output layer. The aggregation function is the so-called *maximum*, and the jumping knowledge mode is the so-called *lstm*. The learning rate parameter is fixed at a value of 0.001.

Figure 15(a) shows the loss as a function of epochs for the training dataset, and Fig. 15(b) shows the performance metrics (such as mean absolute error (mae) and root mean square error (rmse)) for the training, validation, and test datasets. The 95 % confidence intervals for the means are also shown in each plot, calculated using the bootstrapping algorithm [76]. Our results indicate that only a few epochs (typically, 50 epochs) are needed to optimise the parameters of the machine learning model.

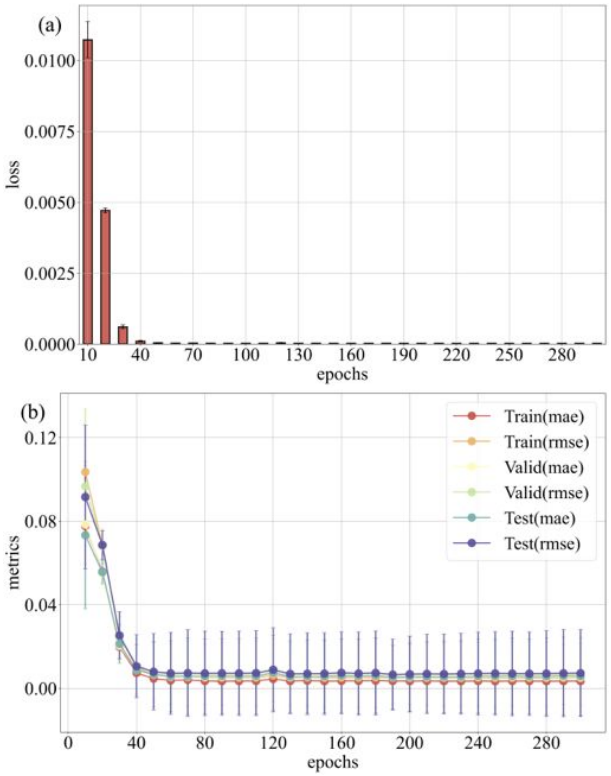

Figure 15: Test system: (a) Loss as a function of epochs for the training dataset; (b) Performance metrics (mean-absolute-error and root-mean-square-error) for the training, validation, and test datasets.

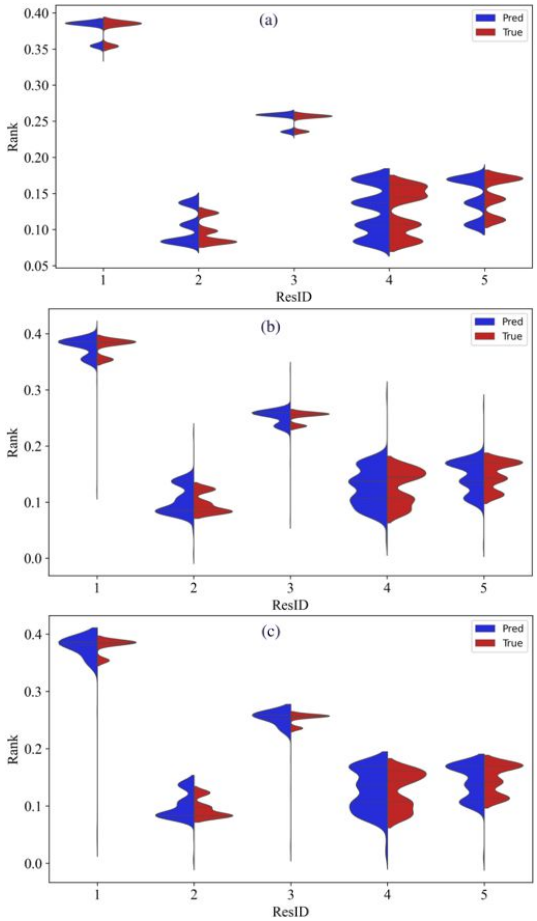

Figure 16: Test system: Predicted and the true (measured) PageRank distributions based on (a) the training dataset, (b) the validation dataset, and (c) the test dataset.

Furthermore, in Fig. 16(a)-(c), we present the predicted and the actual PageRank distributions of the training, validation, and test datasets. The results indicate excellent agreement between the predicted and true distributions for the three datasets.

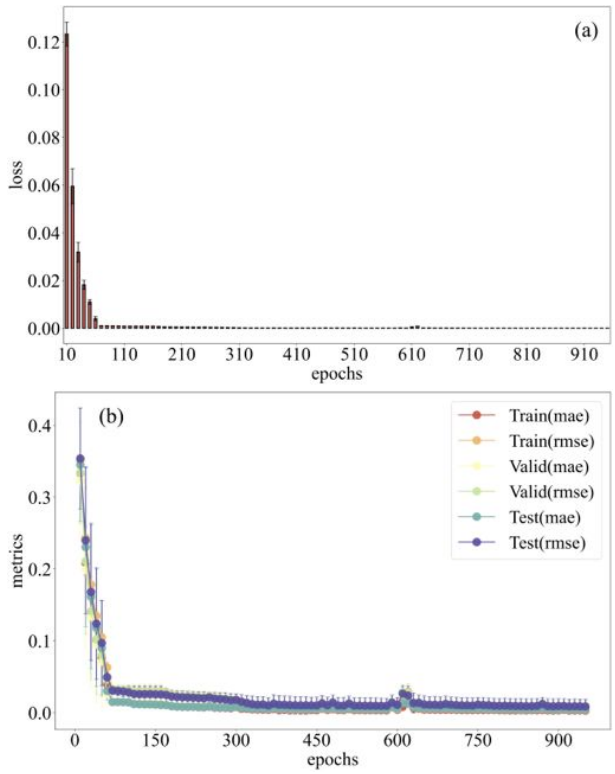

Figure 17: BPTI[14-38]Abu protein: (a) Loss as a function of epochs for the training dataset; (b) Performance metrics (mean-absolute-error and root-mean-square-error) for the training, validation, and test datasets.

4.6.4. BPTI[14-38]Abu Protein

Next, we considered the biological system, BPTI[14-38]Abu protein. In this study, the Tsallis local transfer entropy matrix $t^{(q)}_{i\to j}(t)$ between the amino acids i and j is obtained from short MD simulations data ($t = 0, 1, \cdots, T-1$). We used the kernel density estimator approach with $q = 0.85$, as described in Supporting

Information. The dataset size is $T = 1000$ frames, and the training, validation, and test dataset lengths are 80 %, 10 %, and 10 %, respectively. The batch sizes are $(100, 1, 1)$ and several epochs of 1000. Here, the number of layers is ten, and the number of channels at each layer is 16. Again, the *adam* optimiser is used with a *relu* activation function for the hidden layers and the sigmoid function for the output layer. The aggregation function again was *maximum* and the *lstm* jumping knowledge mode. The learning rate parameter is 0.001.

Figures 17(a)-(b) present the loss as a function of epochs for the training dataset and the performance metrics for the training, validation, and test datasets. Again, the results show that in about 50-100 epochs the parameters of the computational model are optimised.

In Fig. 18(a)-(c), we also present the predicted and the measured (true) PageRank values based on the training, validation, and test datasets, which again show good agreement between the predicted and true distributions in all datasets.

Our results indicate that the DGCNN model can predict the distributions of the PageRank values for all datasets, including the test dataset. Note that computations of the local transfer entropies can be costly; therefore, developing predictive machine learning models of the PageRank distributions relying on short length measurements can reduce significantly the computation demands.

5. Conclusions

This study discussed some alternative definitions for the information-theoretic measures of transfer entropy between time series using (α, q)-framework measures to validate statistical efficiency. The primary goal was to develop new measures of high-order correlations that preserved information-theoretic properties of information transfer due to local operations.

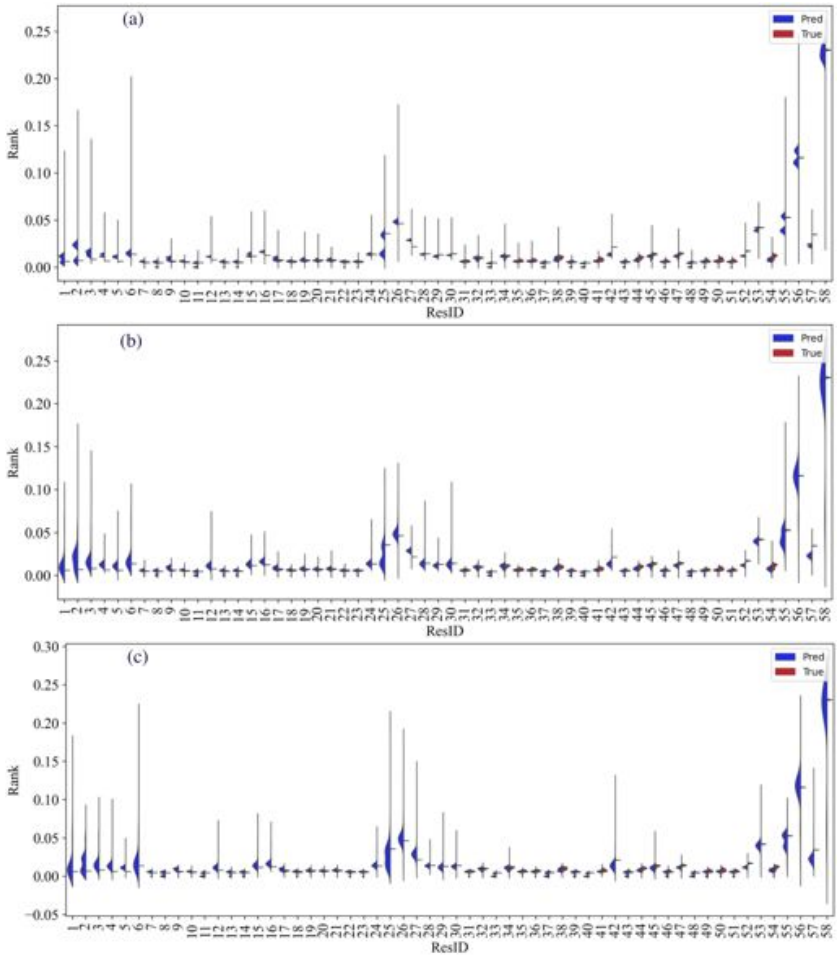

Figure 18: BPTI[14-38]Abu protein:: Predicted and the true (measured) PageRank distributions based on (a) the training dataset; (b) the validation dataset; and (c) the test dataset.

Besides, this study revealed the decoded information contained in an amino acid sequence establishing a 3D protein structure by comparing the physical-chemical properties with the ranked role in the protein interaction network topology using new graph-theoretic measures based on the constructed DiGraph models of (α, q) transfer entropy.

Moreover, in this study, the DGCNN model was used to classify the amino acid influence in a protein structure trained upon short equilibrium structure fluctuations at sub-nanosecond time scales, as obtained from MD simulations. For that, a heat flow kernel DiGraph embedding theoretical framework was developed. In this machine learning computational model, the amino acid information (represented by the vertices of a directed graph) is encoded by an Euclidean vector space embedded vector. It may be interesting to explore the mapping into other vector spaces using the differential geometry of a graph structure.

We examined the influence of disulphide bridges on the 3D structure of the BPTI wild- and mutated-type proteins.

Acknowledgments

The author is grateful to the International Balkan University supporting the research activity.

Data Availability

The supplementary materials used in this study are available within the article and its Supporting Information. The molecular dynamics simulations data used in this study are available from the corresponding author upon request.

References

[1] M. Karplus, J. McCammon, Molecular dynamics simulations of biomolecules, Nat. Struct. Biol. 9 (10) (2002) 646–652.

[2] T. Lazaridis, M. Karplus, Effective energy function for proteins in solution, Proteins 35 (1999) 133–152.

[3] W.F. van Gunsteren, D. Bakowies, R. Baron, I. Chandrasekhar, M. Christen, X. Daura, P. Gee, D.P. Geerke, A. Glättli, P.H. Hünenberger, M.A. Kastenholz, C. Oostenbrink, M. Schenk, D. Trzesniak, N.F.A. van der Vegt, H.B. Yu, Biomolecular Modeling: Goals, Problems, Perspectives, Angew. Chem. Int. Ed. 45 (25) (2006) 4064–4092.

[4] M. C. Thomas, A. T. Joy, Elements of information theory, John Wiley & Sons, Inc., Hoboken, 2006.

[5] O. Lange, H. Grubmüller, Generalized correlation for biomolecular dynamics, Proteins 62 (2006) 1053–1061.

[6] T. Ichiye, M. Karplus, Collective motions in proteins: a covariance analysis of atomic fluctuations in molecular-dynamics and normal mode simulations, Proteins 11 (1991) 205–217.

[7] H. Kamberaj, A. van der Vaart, An optimised replica exchange method for molecular dynamics simulations, J. Chem. Phys. 130 (2009) 074904.

[8] H. Kamberaj, A. der Vaart, Extracting the causality of correlated motions from molecular dynamic simulations, Biophys. J. 97 (2009) 1747–1755.

[9] T. Schreiber, Measuring Information Transfer, Phys. Rev. Lett. 85 (2000) 461–464.

[10] X. S. Liang, R. Kleeman, Information transfer between dynamical system components, Phys. Rev. Lett. 95 (2005) 244101.

[11] B. Gourévitch, J. Eggermont, Evaluating information transfer between auditory cortical neurons, J. Neurophysiol. 97 (2007) 2533–2543.

[12] M. Staniek, K. Lehnertz, Symbolic transfer entropy, Phys. Rev. Lett. 100 (2008) 158101.

[13] M. Prokopenko, J. T. Lizier, Transfer entropy and transient limits of computation, Sci. Rep. 4 (2014) 5394.

[14] J. Pahle, A. K. Green, C. J. Dixon, U. Kummer, Information transfer in signaling pathways: A study using coupled simulated and experimental data, BMC Bioinformatics 9 (2008) 139.

[15] K. Kriefall, D. Barr, Drug binding alters pathways of information flow in ERK2, Biophys. J. 121 (3) (2022) 340a.

[16] L. G. Michel, C. Keirns, B. Ahlbrecht, D. Barr, Calculating Transfer Entropy from Variance-Covariance Matrices Provides Insight into Allosteric Communication in ERK2, J. Chem. Theory Comput. 17 (5) (2021) 3168–3177.

[17] K. Kriefall, K. J. Oweis, D. Barr, Transfer entropy and protein allostery: promise, limits, and applications, Biophys. J. 121 (3) (2022) 133a–134a.

[18] A. Hacisuleyman, B. Erman, Causality, Transfer Entropy and Allosteric Communication Landscapes in Proteins with Harmonic Interactions, Proteins: Structure, Function, and Bioinformatics 85 (6) (2017) 1056–1064.

[19] A. Hacisuleyman, B. Erman, Entropy Transfer between Residue Pairs and Allostery in Proteins: Quantifying Allosteric Communication in Ubiquitin, PLoS Comput. Biol. 13 (1) (2017) e1005319.

[20] H. Kamberaj, Heat flow random walks in biomolecular systems using symbolic transfer entropy and graph theory, J. Mol. Graph. Model. 104 (2021) 107838.

[21] V. Ilić, I. B. Djordjević, On the $\alpha - q$-mutual information and the $\alpha - q$-capacities, Entropy 23 (2021) 702–725.

[22] S. O. Scalet, A. M. Alhambra, G. Styliaris, J. I. Cirac, Computable Rényi mutual information: Area laws and correlations, Quantum 5 (2021) 541.

[23] Dh. Nebiu, H. Kamberaj, Symbolic Information Flow Measurement (SIFM): A software for measurement of information flow using symbolic analysis, SoftwareX 11 (2020) 100470.

[24] J. Runge, V. Petoukhov, J. F. Donges, J. Hlinka, N. Jajcay, M. Vejmelka, D. Hartman, N. Marwan, M. Palus, J. Kurths, Identifying causal gateways and mediators in complex spatio-temporal systems, Nature Communications 6 (2015) 8502.

[25] S. J. D. Prince, Understanding Deep Learning, MIT Press, 2023.

[26] H. ElGhawalby, E. R. Hancock, Heat Kernel Embeddings, Differential Geometry and Graph Structure, Axioms 4 (2015) 275–293.

[27] S. Mousazadeh, I. Cohen, Embedding and function extension on directed graph, Signal Processing 111 (2015) 137–149.

[28] A. Wlodawer, J. Walter, R. Huber, L. Sjolin, Structure of bovine pancreatic trypsin inhibitor. Results of joint neutron and X-ray refinement of crystal form II, J. Mol. Biol. 180 (1984) 301–329.

[29] I. V. Getun, C. K. Brown, J. Tulla-Puche, D. Ohlendorf, C. Woodward, G. Barany, Partially folded Bovine Pancreatic Trypsin Inhibitor analogues attain fully native structures when co-crystallized with S195A rat trypsin, J. Mol. Biol. 375 (2008) 812.

[30] B. Sharma, D. Mittal, New non-additive measures of entropy for discrete probability distributions, Journal of Mathematical Sciences 10 (1975) 28–40.

[31] A. Rényi, Contributions to the Theory of Statistics, in: Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, University of California Press, 1961, pp. 547–561.

[32] C. Tsallis, Possible generalization of Boltzmann-Gibbs statistics, J. Stat. Phys. 52 (1988) 479–487.

[33] T. Yamano, A Possible Extension of Shannon's Information Theory, Entropy 3 (2001) 280–292.

[34] H. Kamberaj, Random walks in a free energy landscape combining augmented molecular dynamics simulations with a dynamic graph neural network model, J. Mol. Graph. Model. 114 (2022) 108199.

[35] S. Behrendt, T. Dimpfl, F. J. Peter, D. J. Zimmermann, RTransferEntropy − Quantifying information flow between different time series using effective transfer entropy, SoftwareX 10 (2019) 100265.

[36] J. T. Lizier, M. Prokopenko, A. Y. Zomaya, Local information transfer as a spatiotemporal filter for complex systems, Phys. Rev. E 77 (2008) 026110.

[37] C. Tsallis, Comment on "critique of q-entropy for thermal statistics", Phys. Rev. E 69 (2004) 038101–6.

[38] C. Tsallis, The Nonadditive Entropy S_q and Its Applications in Physics and Elsewhere: Some Remarks, Entropy 13 (2011) 1765–1804.

[39] K. P. Nelson, Open Problems within Nonextensive Statistical Mechanics, Entropy 26 (2024) 118.

[40] T. D. Schneider, Theory of molecular machines. II. Energy dissipation from molecular machines, J. Theor. Biol. 148 (1991) 125–137.

[41] M. Prokopenko, J. T. Lizier, D. C. Price, On thermodynamic interpretation of transfer entropy, Entropy 15 (2013) 524–543.

[42] H. Kamberaj, Molecular Dynamics Simulations in Statistical Physics. Theory and Applications, Scientific Computation Series, Springer Nature, 2020.

[43] W. Humphrey, A. Dalke, K. Schulten, VMD-Visual Molecular Dynamics, J. Mol. Graph. 14 (1996) 33–38.

[44] B. R. Brooks, C. L. Brooks, A. D. MacKerell, L. Nilsson, R. J. Petrella, B. Roux, Y. Won, G. Archontis, C. Bartels, S. Boresch, A. Caflisch, L. Caves, Q. Cui, A. R. Dinner, M. Feig, S. Fischer, J. Gao, M. Hodoscek, W. Im, K. Kuczera, T. Lazaridis, J. Ma, V. Ovchinnikov, E. Paci, R. W. Pastor, C. B. Post, J. Z. Pu, M. Schaefer, B. Tidor, R. M. Venable, H. L. Woodcock, X. Wu, W. Yang, D. M. York, M. Karplus, CHARMM: The biomolecular simulation program, J. Comput. Chem. 30 (10) (2009) 1545–1614.

[45] J. A. D. MacKerell, D. Bashford, M. Bellott, Jr. R. L. Dunbrack, J. D. Evanseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kuchnir, K. Kuczera, F. T. K. Lau, C. Mattos, S. Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. Reiher, B. Roux, M. Schlenkrich, J. C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wiorkiewicz-Kuczera, D. Yin, M. Karplus, All-atom empirical potential for molecular modeling and dynamics studies of proteins, J. Phys. Chem. B 102 (1998) 3586–3616.

[46] A. D. Jr. MacKerell, M. Feig, C. L. Brooks III, Improved treatment of the protein backbone in empirical force fields, J. Am. Chem. Soc. 126 (2004) 698–699.

[47] B. Robert, R. B. Best, X. Zhu, J. Shim, P. Lopes, J. Mittal, M. Feig, A. D. Jr. MacKerell, Optimization of the additive CHARMM all-atom protein force field targeting improved sampling of the backbone Φ, Ψ and sidechain χ_1 and χ_2 dihedral angles, J. Chem. Theory Comput. 8 (2013) 3257–3273.

[48] J. C. Gordon, J. B. Myers, T. Folta, V. Shoja, L. S. Heath, A. Onufriev,

H++: A server for estimating pKa and adding missing hydrogens to macromolecules, Nucleic Acids Research 33 (Web server issue) (2005) W368–W371.

[49] W. Jorgensen, J. Chandrasekhar, R. J. Madura, M. Klein, Comparison of simple potential functions for simulating liquid water, J. Chem. Phys. 79 (1983) 926–935.

[50] H. Kamberaj, Computer Simulations in Molecular Biology, Scientific Computation Series, Springer Nature, 2023.

[51] J. P. Ryckaert, G. Ciccotti, H. J. C. Berendsen, Numerical integration of the Cartesian equations of motion of a system with constraints: Molecular dynamics of n-alkanes, J. Comput. Phys. 23 (1977) 327–341.

[52] M.P. Allen, D.J. Tildesley, Computer simulation of liquids, Oxford University Press, 1989.

[53] F. Blomberg, W. Maurer, H. Rüterjans, ^{15}N nuclear magnetic resonance investigations on amino acids, Proc. Natl. Acad. Sci. USA 73 (5) (1976) 1409–1413.

[54] R. M. Levy, M. Karplus, Trajectory studies of NMR relaxation in flexible molecules, American Chemical Society, 1983.

[55] E. Barbar, G. Barany, C. Woodward, Unfolded BPTI variants with a single disulfide bond have diminished non-native structure distant from the crosslink, Folding and Design 1 (1) (1996) 65–76.

[56] Y. Kuroda, P. S. Kim, Folding of Bovine Pancreatic Trypsin Inhibitor (BPTI) Variants in which Almost Half the Residues are Alanine, J. Mol. Biol. 298 (2000) 493–501.

[57] T. Lazaridis, M. Karplus, Thermodynamics of protein folding: a microscopic view, Biophysical Chemistry 100 (2003) 367–395.

[58] R. Izairi, H. Kamberaj, Comparison Study of Polar and Non - polar Contributions to Solvation Free Energy, Journal of Chemical Information and Modeling 57 (10) (2017) 2539–2553.

[59] M. Gurboga, G. Kugu, H. Kamberaj, O. Mutlu, Identification of Benzamide Inhibitors for Histone Deacetylase 1 from Babesia and Theileria Species by Virtual High-Throughput Screening and Molecular Dynamics Simulations, Parasitology Research 120 (2021) 2175–2187.

[60] M. Karplus, J. Jushick, Method for estimating the configurational entropy of macromolecules, Macromolecules 14 (1981) 325–332.

[61] I. Andricioaei, M. Karplus, On the calculation of entropy from covariance matrices of the atomic fluctuations, J. Chem. Phys. 115 (2001) 6289–6292.

[62] J. Chen, C. L. Brooks III, Implicit modelling of non polar solvation for simulating protein folding and conformational transitions, Phys. Chem. Chem. Phys. 10 (2008) 471–481.

[63] J. Chen, C. L. Brooks III, Implicit modelling of non polar solvation for simulating protein folding and conformational transitions, Phys. Chem. Chem. Phys. 10 (2008) 471–481.

[64] F. M. Dekking, C. Kraaikamp, H. P. Lopuhaa, L. E. Meester, A modern introduction to probability and statistics, Springer-Verlag, 2005.

[65] P. Bonacich, A Family of Measures, American Journal of Sociology 92 (5) (1986) 1170–1182.

[66] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank citation ranking: Bringing order to the Web, Unpublished manuscript.
URL `google.stanford.edu/~backrub/pageranksub.ps`

[67] U. Brandes, A faster algorithm for betweenness centrality, J. Math. Soc. 25 (2001) 163–177.

[68] S. Chakrabarti, Mining the Web. Discovering Knowledge from Hypertext Data, Morgan Kaufmann Publishers, San Francisco, 2003.

[69] G. Fagiolo, Clustering in complex directed networks, Phys. Rev. E 76 (2007) 026107.

[70] M. E. J. Newman, Networks: An Introduction, Oxford University Press, 2010.

[71] J. Y. Yen, Finding the K Shortest Loopless Paths in a Network, Management Science 17 (11) (1971) 712–716, theory Series.

[72] R. A. Ramadan, A. Y. Khedr, K. Yadav, E. J. Alreshidi, Md H. Sharif, A. T. Azar, H. Kamberaj, Convolution neural network based automatic localization of landmarks on lateral X-ray images, Multimedia Tools and Applications 81 (2022) 37403.

[73] K. Xu, C. Li, Y. Tian, T. Sonobe, K. -i. Kawarabayashi, S. Jegelka, Representation learning on graphs with jumping knowledge networks, International Conference on Machine Learning 267 (2018) 5453–5462.

[74] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (8) (1997) 1735–1780.

[75] G. Li, M. Müller, G. Qian, I. C. Delgadillo, A. Abualshour, A. Thabet, B. Ghanem, DeepGCNs: Can GCNs Go as Deep as CNNs?, IEEE Transactions on Pattern Analysis and Machine Intelligence 45 (6) (2023) 6923–6930.

[76] M. Hoxha, H. Kamberaj, Automation of Some Macromolecular Properties Using a Machine Learning Approach, Machine Learning: Science and Technology 2 (2021) 035016.