

SI METHODS

Identifying Potential Allosteric Residues

Identifying Surface-Critical Residues

All biological assembly files were downloaded from the Protein Data Bank (26). With the objective of identifying potential allosteric residues on the protein surface, we employed a modified version of the binding leverage method for identifying likely ligand binding sites (Fig. 1A), as described previously (14). Allosteric signals may be transmitted over large distances by a mechanism in which the allosteric ligand has a global affect on a protein's functionally important motions. For instance, introducing a bulky ligand into the site of an open pocket may disrupt large-scale motions if those motions normally entail that the pocket become collapsed over the course of a motion. Such a modulation of the global motions may affect activity within sites that are distant from the allosteric ligand-binding site.

We refer the reader to the work by Mitternacht and Berezovsky for details regarding the binding leverage method, though a general overview of the approach follows (14). Many candidate allosteric sites are generated by simulations in which a simple flexible ligand (comprising of 4 “atoms” linked by bonds of fixed length 3.8 Angstroms, but variable bond and dihedral angles) explores the protein's surface through many Monte Carlo steps. The number of Monte Carlo simulations is set to 10 times the number of residues in the protein structure, and the number of MC steps per simulation is set to 10,000 times the size of the simulation box (in our implementation), as measured in Angstroms. The size of this simulation box is set to twice the maximum size of the PDB along any of the x, y or z-axes. *Apo* structures were used when probing protein surfaces for putative ligand binding sites in the canonical set of proteins.

A simple square well potential (i.e., modeling hard-sphere interactions) is used to model the attractive and repulsive energy terms associated with the ligand's interaction with the protein surface. In the unmodified implementation of the method, these energy terms depend only on the ligand atoms' distance to *alpha carbon atoms* in the protein – other heavy atoms or biophysical properties are not considered.

Once these candidate sites are produced, normal modes analysis is applied to generate a model of the protein's low-frequency motions. To generate these modes, we use the alpha carbon atoms in building the protein's elastic networks. Using default parameters, we use the top 10 (lowest-frequency) available non-trivial modes generated using the Molecular Modeling Toolkit (MMTK) (39). Note that this exact same method for producing the modes was also used in the identification of interior-critical residues (below).

Once the modes are produced, each of the candidate sites is then scored based on the degree to which deformations in the site couple to the low-frequency modes; that is, those sites which are heavily deformed as a result of the normal mode fluctuations (Fig. 1A, top-right) receive a high score (termed the binding leverage for that site), whereas shallow sites with few interacting residues (Fig. 1A, bottom-left) or sites that undergo minimal change over the course of a mode fluctuation (Fig. 1A, bottom-right) receive a low binding leverage score. Strongly overlapping sites are merged, and the list is then ranked by binding leverage score. This generates a ranked list of N sites. Using knowledge of the experimentally determined binding sites (i.e., from *holo* structures), the processed list of ranked sites is then used to evaluate predictive performance (see below).

Our approach and set of applications differ from those previously developed in several key ways. When running Monte Carlo simulations to probe the protein surface and generate candidate binding sites, we used all heavy atoms in the protein when evaluating a ligand's affinity for each location. By including all heavy atoms (i.e., as oppose to using the protein's alpha carbon atoms exclusively), our hope is to generate a more selective set of candidate sites. Indeed, the use of alpha carbon atoms alone leaves 'holes' in the protein which do not actually exist in the context of the dense topology of side chain atoms. Thus, by including all heavy atoms, we hope to reduce the number of false positive candidate sites, as well as more realistically model ligand binding affinities in general.

In the original binding leverage framework, an interaction between a ligand atom and an alpha carbon atom in the protein contributes -0.75 to the binding energy if the interaction distance is within the range of 5.5 to 8 Angstroms. Interaction distances greater than 8 Angstroms do not contribute to the binding energy, but distances in the

range of 5.0 to 5.5 are repulsive, and those between 4.5 to 5.0 Angstroms are strongly repulsive (distances below 4.5 Angstroms are not permitted). However, given the much higher density of atoms interacting with the ligand in our all-heavy atom model of each protein, it is necessary to accordingly change the energy parameters associated with the ligand's binding affinity. In particular, we varied both the ranges of favorable and unfavorable interactions, as well as the attractive and repulsive energies themselves. That is, we varied both the square well's width as well as depth when evaluating the ligand's affinity for a given site.

For well depths, we employed models using attractive potentials ranging from -0.05 to -0.75, including all intermediate factors of 0.05. For well widths, we tried performing the ligand simulations using the cutoff distances originally used (attractive in the range of 5.5 to 8.0 Angstroms, repulsive in the range of 5.0 to 5.5, and strongly repulsive in the range of 4.5 to 5.0). However, these cutoffs, which were originally devised to model the ligand's affinity to the alpha carbon atom skeleton alone, were observed to be inappropriate when including all heavy atoms. Thus, we also performed the simulations using a revised set of cutoffs, with attractive interactions in the range of 3.5 to 4.5 Angstroms, repulsive interactions in the range of 3.0 to 3.5 Angstroms, and strongly repulsive interactions in the range of 2.5 to 3.0 Angstroms.

In order to identify the optimal set of parameters for defining the potential function, we determined which combination of parameters best predicts known ligand binding sites in threonine synthase (1E5X), phosphoribosyltransferase (1XTT), tyrosine phosphatase (2HNP), arginine kinase (3JU5), and adenylate kinase (4AKE). Using this approach, an attractive term of -0.35 for ligand-protein atom interactions within the range of 3.5 to 4.5 Angstroms was determined to be the best overall.

As discussed in the main text, without applying thresholding to the list of ranked surface sites that remain after running the binding leverage calculations, a very large number of sites would occupy the protein surface (Fig. S2). Thus, it is necessary to process this list. To do so, we borrow from principles in energy gap theory (40). The calculations for establishing a threshold on the number of sites are as follows:

For each of the N candidate binding sites in the pre-processed ranked list of sites, we calculate $\partial \text{BL}(j) / \Delta \text{BL}$. Here, j is the j^{th} site to appear in the pre-processed ranked

list of sites, with this list of sites being ranked on the basis of each site's binding leverage score (see above). $\partial\text{BL}(j)$ is defined as the difference in the binding leverage scores of the j^{th} site and the $(j-1)^{\text{th}}$ site in the ranked list. Because the list of sites is organized in descending order of binding leverage scores, $\partial\text{BL}(j) \geq 0$. ΔBL is a constant that equals $\max_{\text{binding_leverage_score}} - \min_{\text{binding_leverage_score}}$ in the pre-processed ranked list of sites. ΔBL is thus the top binding leverage score that appears in this ranked list minus the bottom score. Qualitatively, a large value for $\partial\text{BL}(j) / \Delta\text{BL}$ indicates that there is a large drop in binding leverage scores in going from site (j) to site $(j-1)$ within the pre-processed ranked list.

We then consider those sites with the highest $\partial\text{BL} / \Delta\text{BL}$ values – specifically, we consider the top 5.5% of sites in terms of their $\partial\text{BL} / \Delta\text{BL}$ values. Thus, we are considering site j if there is a very large gap in binding leverage scores between sites j and $(j-1)$. The lowest-occurring site within *this* considered list of high $\partial\text{BL} / \Delta\text{BL}$ values demarcates a threshold beyond which we reject all lower sites within the pre-processed ranked list, leaving only the processed ranked list of sites.

We then go up from bottom through the top of this processed ranked list of sites, and for each site, we determine the jaccard similarity with all sites above. If the jaccard similarity with any site above exceeds 0.7, then the lower site is removed from the processed ranked list. The final list obtained after performing these jaccard similarity filters is taken to represent the set of surface-critical sites on a structure.

In counting the final number of truly *distinct* surface-critical sites for any given structure, we remove redundant sites within the set of surface-critical sites obtained in the process above, as some of the sites within this set may still exhibit considerable mutual overlap. A site i within the list of surface-critical sites is said to be redundant if it is assigned a redundancy score that exceeds 0.4, where $\text{redundancy_score}(i) = [\text{residues}_{\text{site } i} \cap \{ \cup (\text{set of residues in all accepted sites above site } i \text{ in the ranked list of sites}) \}] / [\# \text{ residues in site } i]$. If this redundancy score is less than 0.4, then site i is included in the list of accepted sites. If it exceeds 0.4, then the site overlaps with another site on the surface, and it is thus rejected from the set of accepted distinct sites. Finally, the total number of sites in the accepted set of sites is taken as the number of distinct sites for a structure.

Capturing Known Ligand-Binding Sites

Known ligand-binding residues are those within 4.5 Angstroms of the ligand in the *holo* structure (Table S1). It has previously been shown that the sites in aspartate transcarbamoylase are especially difficult to identify (14); excluding this protein results in finding an average of 65% of known biological ligand binding sites (Table S2). Note that these statistics are achieved by covering an average of 15% of proteins' residues, even though more than 15% of residues are actually involved in ligand- or substrate-binding for most proteins (Table S3).

Dynamical Network Analysis to Identify Interior-Critical Residues

In our implementation of the Girvan-Newman framework (Fig. 1B), an edge between residues i and j is drawn if any heavy atom within residue i is located within 4.5 Angstroms of any heavy atom within residue j , and we exclude the trivial cases of pairs of residues that are adjacent in sequence, which are not considered to be in contact.

Network edges are then weighted on the basis of correlated motions of the interacting residues, with these motions provided by the same ANMs that had been used in the identification of surface-critical residues. We emphasize that, although ANMs are more coarse-grained than molecular dynamics, our use of ANMs is motivated by their much faster computational efficiency. This added efficiency is a required feature for our database-scale analysis. As an alternative to using ANMs, it is also possible to infer motion by simply using information regarding pairs of distinct conformations (see SI Methods subsection titled “Modeling Protein Motions by Directly Using Displacement Vectors from Alternative Conformations”, below).

The edge weighting scheme is performed as follows: an “effective distance” d_{ij} for an edge between interacting residues i and j is set to $d_{ij} = -\log(|C_{ij}|)$, where C_{ij} designates the correlated motions between residue i and j :

$$C_{ij} = Cov_{ij} / \sqrt{(\langle \mathbf{r}_i^2 \rangle \langle \mathbf{r}_j^2 \rangle)}$$

where

$$Cov_{ij} = \langle \mathbf{r}_i \cdot \mathbf{r}_j \rangle$$

Here, \mathbf{r}_i and \mathbf{r}_j designate the vectors associated with residues i and j (respectively) under a particular mode. The brackets in the term $\langle \mathbf{r}_i \cdot \mathbf{r}_j \rangle$ indicate that we are taking the mean value for the dot product $\mathbf{r}_i \cdot \mathbf{r}_j$ over the 10 lowest-frequency non-trivial modes.

An example may help to clarify this. If two interacting residues exhibit a *high* degree of correlated motion, then the motion of one may tell us about the motion of the other, suggesting a strong flow of energy or information between the two residues, resulting in a *low* value for d_{ij} : a strong correlation (or a strong anti-correlation) between nodes i and j result in a value for $|C_{ij}|$ that is close to 1. This gives a low value for d_{ij} ($-\log(|C_{ij}|) \approx 0$). Thus, given a strong correlated motion, this effective distance d_{ij} between residues i and j is very short. This small d_{ij} means that any path involving this pair of residues is likewise shorter as a result, thereby more likely placing this pair of residues within a shortest path, and more likely rendering this pair a bottleneck pair. In sum, this edge-weighting scheme is such that a high correlation in motion results in a short effective distance, thereby more likely rendering this a bottleneck pair of residues.

In the opposite scenario, for interacting residues with poor correlation values ($C_{ij} \approx 0$), a large effective distance d_{ij} results, thereby making it more difficult for the pair of residues to lie within shortest paths or within the same community.

Once all connections between interacting pairs of residues are appropriately weighted and the communities are assigned using the GN algorithm with these effective distances, a residue is deemed to be critical for allosteric signal transmission (i.e., an interior-critical residue) if it is involved in the highest-betweenness edge connecting two distinct communities. A given edge's betweenness is taken to be the number of shortest paths involving that edge. For demonstration, this results in the community partitions and associated interior-critical residues highlighted in Figs. S3 and S4.

Decomposing Proteins into Modules Using Different Algorithms

Many algorithms have been devised to identify the community structure of networks. By this, we are referring to the problem of finding the optimal partitioning of a network into different “modules” (i.e., communities), such that each node within a module is highly connected to other nodes within the same module, and minimally connected to other nodes in outside modules. In a comprehensive study comparing different algorithms

(41), an information theory-based approach (42) was shown to be one of the strongest. This method (termed “Infomap”) effectively reduces the network community detection problem to a problem in information compression: the prominent features of the network are extracted in this compression process, giving rise to distinct modules; further details are provided in (42).

Perhaps surprisingly, even though both Infomap and GN achieve similar network modularity, Infomap produces at least twice the number of communities relative to that of GN when applied to protein structures, and it thus generates many more interior-critical residues (Table S5 and Fig. S20). Within the set of 12 canonical proteins, GN and Infomap generate an average of 12.0 and 36.8 communities, respectively. This corresponds to an average of 44.8 and 201.4 interior-critical residues when using GN and Infomap, respectively. Thus, given that GN produces a more selective set of residues for each protein, we use GN throughout our analyses.

Although the critical residues identified by GN do not always correspond to those identified by Infomap, the mean fraction of GN-identified interior-critical residues that match Infomap-identified residues is 0.30 (the expected mean, based on a uniformly-random distribution of critical residues throughout the protein, is 0.21, p -value=0.058), further justifying our decision to focus on GN). Furthermore, we observe that obvious structural communities are detected when applying both methods: a community generated by GN is often the same as that generated by Infomap, and in other cases, a community generated by GN is often composed of sub-communities generated by Infomap.

As noted, the modularity from the network partitions generated by GN and Infomap are very similar. For the 12 canonical systems, the mean modularity for GN and Infomap is 0.73 and 0.68, respectively. Presumably, GN modularity values are consistently at least as high as those in Infomap because GN explicitly optimizes modularity in partitioning the network, whereas Infomap does not.

STRESS (STRucturally-identified ESSential residues)

Our server has been designed to be both user-friendly and highly efficient. We use local searching supported by hashing to perform a local search in each sampling step of the Monte Carlo simulations, which takes constant time. This approach brings down the

asymptotic computational complexity by an order of magnitude, relative to a simpler implementation without optimization (Fig. S6). The time complexity of the core computation, Monte Carlo sampling, is $O(T|S|)$, where T and S are simulation trials and steps for each trial, respectively. After carefully profiling and optimizing for speed (with optimizations introduced through changes in the workflow, data structures, numerical arithmetic, etc.), a typical case takes ~30 minutes on a E5-2660 v3 (2.60GHz) core.

In terms of operation, our tool utilizes two types of servers: front-end servers that handle incoming HTTP requests and back-end servers that perform algorithmic calculations (Fig. S7). Communication between these two types of servers is handled by Amazon's Simple Queue Service. When our front-end servers receive a new request, they add the job to the queue and then return to requests immediately. Our back-end servers poll the queue for new jobs and run them when capacity is available. Amazon's Elastic Beanstalk offers several features that enable us to dynamically scale our web application. We use Auto Scaling to automatically adjust the number of back-end servers backing our application based on predefined conditions, such as the number of jobs in the queue and CPU utilization. Elastic Load Balancer automatically distributes incoming network traffic. This system ensures that we are able to handle varying levels of demand in a reliable and cost-effective manner. Since we may have multiple servers backing our tool simultaneously (some handling HTTP requests and some performing calculations, any of which may be terminated at any time by Auto Scaling), it is important that our servers are stateless. We thus store input and output files remotely in an S3 bucket, which is accessible to each server via RESTful conventions. The corresponding source code and README files are made available through Github (github.com/gersteinlab/STRESS).

High-Throughput Identification of Alternative Conformations

An overview of our pipeline for identifying alternative conformations is provided in Fig. S9. We perform MSAs for thousands of structures, with each alignment consisting of sequence-identical groups. Within each alignment, we cluster structures using RMSD

to determine the distinct conformational states. We then use information regarding protein motions to identify surface- and interior-critical residues. Details are given below.

Database-Wide Multiple Structure Alignments

FASTA files of all SCOP domains were downloaded from the SCOP website (version 2.03) (43, 44). We first worked with domains to probe for intra-domain conformational changes, as better alignments are generally possible at the domain level.

In order to better ensure that large structural differences between domains are a result of differing biological states (such as *holo* vs. *apo*, phosphorylated vs. unphosphorylated, etc.), and not an artifact of missing coordinates in X-ray crystal structures, the FASTA sequences used were those corresponding to the ATOM records of their respective PDBs. In total, this set comprises 162,517 FASTA sequences.

BLASTClust (45) was used to organize these FASTA sequences into sequence-similar groups at seven levels of sequence identity (100%, 95%, 90%, 70%, 50%, 40%, and 30%). Thus, for instance, running BLASTClust with a parameter value of 100 provides a list of FASTA sequence groups such that each sequence within each group is 100% sequence identical, and in general, running BLASTClust with any given parameter value provides sequence groups such that each member within a group shares at least that specified degree of sequence identity with any other member of the same group (see top of Fig. 1). Note that sequence identity values below 100% were only used to evaluate the pairwise RMSD distributions shown in Fig. S21. For all other analyses reported, all results are based on groups of structures that are 100% sequence identical.

To ensure that the X-Ray structures used in our downstream analysis are of sufficiently high quality, we removed all of those structures corresponding to PDB files with resolution values poorer than 2.8, as well as any PDB files with R-Free values poorer than 0.28. The question of how to set these quality thresholds is an important consideration, and was guided here by a combination of the thresholds conventionally used in other studies which rely on large datasets of structures, as well as the consideration that many interesting allosteric-related conformational changes may correlate with physical properties that sometimes render very high resolution values difficult (such as localized disorder or order-disorder transitions). As a result of applying

these filters, 45,937 PDB IDs out of a total of 58,308 unique X-Ray structures (~79%) were kept for downstream analysis. The PDB was accessed in December 2013.

For each sequence-similar group at each of the seven levels of sequence identity, we performed multiple structure alignment (MSA) using only those domain structures that satisfy the criteria outlined above. Thus, the MSAs were generated only for those groups containing a minimum of two domains that pass the filtering criteria. The STAMP (46) and MultiSeq (47) plugins of VMD (48) were used to generate the MSAs.

Heteroatoms were removed from each structure prior to performing the alignments.

The quality of the resultant MSA for each sequence-similar group depends on the root structure used in the alignment. To obtain the optimal MSA for each group of N structures, we generated N MSAs, with each alignment using a different one of the N domains as the root. The best MSA (as measured by STAMP's *sc* score) was taken as the MSA for that group. Note that, in order to aid in performing the MSAs, MultiSeq was used to generate sequence alignments for each group.

Finally, MultiSeq was used calculate two measures of structural similarity between each pair of domains within a group: RMSD and Q_H . Q_H , an alternative metric to RMSD, quantifies the degree to which residue-residue distances differ between two conformations, and is detailed in (49). For each group of sequence-similar domains, the final output of the structure alignment is a symmetric matrix representing all pairwise RMSD values (as well as a separate matrix representing all pairwise Q_H values) within that group. The matrices for all MSAs are then used as input to the K-means module. PDB-wide MSAs across sequence-similar groups reveal that, in agreement with expectation, average RMSD values increase at lower levels of sequence identity (Fig. S21). Grouping structures within a multiple-structure alignment on the basis of RMSD did not change substantially when grouping structures using Q_H (Fig. S22). Thus, we use RMSD as a similarity metric throughout.

Identifying Distinct Conformations within a Multiple Structure Alignment

For each MSA produced in the previous step (using only sets of sequences that are 100% sequence-identical), the corresponding matrix of pairwise RMSD values describes the degree and nature of structural heterogeneity among the crystal structures

for a particular protein. The objective is to use this data in order to identify the biologically distinct conformations represented by an ensemble of structures. For a particular structure, there may be many available crystal structures. In total, these structures may actually represent only a small number of distinct biological states and conformations. For instance, there may be several crystal structures in which the domain is bound to its cognate ligand, while the remaining structures are in the *apo* state.

Our framework for identifying the number of distinct conformational states in an ensemble of structures relies on a modified version of the K-means clustering algorithm. This modified form of the algorithm is termed K-means clustering with the gap statistic, and it was introduced in (50).

A priori, performing K-means clustering assumes prior knowledge of the number of clusters (i.e., “K”) to describe a dataset. The purpose of K-means clustering with the gap statistic is to identify the optimal number of clusters intrinsic to a complex or noisy set of data points (which lie in N-dimensional space). Given multiple resolved crystal structures for a given domain, this method estimates the number of conformational states represented in the ensemble of structures (with these states likely occupying different wells within the energetic landscape), thereby identifying proteins which are likely to undergo conformational change as part of their functionality.

As a first step toward clustering the structure ensemble represented by an RMSD matrix, it is necessary to convert this RMSD matrix (which explicitly represents only the *relationships* between distinct structures) into a form in which each structure is given its own set of coordinates. This step is taken because the K-means algorithm acts directly on individual data points, rather than the distances between such points. Thus, we use multidimensional scaling (MDS) to convert an N-by-N matrix (which provides all RMSD values between each pair of structures within a group of N structures) into a set of N distinct points, with each point representing a domain in (N-1)-dimensional space (see below). The values of the N-1 coordinates assigned to each of these N points are such that the Euclidean distance between each pair of points is the same as that corresponding pair’s RMSD value in the original matrix.

For an intuition into why N points are mapped to (N-1)-dimensional space, consider an MSA between two structures. The RMSD between these two structures can

be used to map the two structures to one-dimensional space, such that the distance between the points is the RMSD value. Similarly, an MSA of 3 structures may be mapped to 2-dimensional space in such a way that the pairwise distances are preserved; 4 domains may be mapped to 3-dimensional space, etc. The output of this multidimensional scaling is used as input to the K-means clustering with the gap statistic.

We refer the reader to the work by Tibshirani *et al* for details governing how we perform K-means clustering with the gap statistic, as well as more details on the theoretical justifications of this approach (50). However, an overview of the general intuition behind the formalism is provided here.

For the purpose of demonstration, assume that the data takes the form of 60 data points, with each point represented in 2D space. These are represented by the blue points in Fig. S23. Of course, our observed data in the case of MSAs may lie in N-dimensional space, in which case all Euclidean distances are just as easily calculated.

1) Start by assuming that the input data can be represented with K clusters. Perform Lloyd's algorithm (i.e., standard K-means clustering) on the dataset in order to assign each point to one of K clusters. Then, for each cluster k (which contains data points in the set C_k) measure D_k , which describes the ‘density’ of points within cluster k:

$$D_k = \sum_{x_i \in C_k} \sum_{x_j \in C_k} ||x_i - x_j||^2$$

2) Calculate an overall normalized score W to describe how well-clustered the resultant system has become when assigning all 60 data points to the K clusters (n_k denotes the number of points in cluster k):

$$W = \sum_{k=1}^K \frac{1}{2n_k} D_k$$

3) Given our observed data, how well does this number of assigned clusters K actually represent the ‘true’ number of clusters represented by the data, relative to a null model without any apparent clustering? To address this question, produce a null distribution of 60 randomly (i.e., uniformly) distributed data points that lack any clear

clustering (grey points in Fig. S23) such that the randomly-placed points lie within the same bounding box of the observed data (blue points).

4) Repeat step (3) M times, and each time a random null distribution is produced, calculate $W_{null(K)}$ for each distribution (assuming K clusters), just as W is calculated for the observed data. Then calculate the $\text{mean}_M\{\log(W_{null(K)})\}$ for these M null distributions. Intuitively, the value $\text{mean}_M\{\log(W_{null(K)})\}$ measures how well *random* systems (with the same number of data points and within the same variable ranges as the observed data) can be described by K clusters. The M $\log(W_{null(K)})$ values produced by the null models have a standard deviation that is ultimately converted to s_k (see (50) for details):

$$s_k = \sqrt{1 + 1/B} \text{sd}(k)$$

5) Calculate the gap statistic $\delta(K)$, given K clusters – this is a measure of how well our observed data may be described by K clusters relative to null models containing the same number of points and within the same variable ranges. Intuitively, a high value for this statistic signifies that our data is well-described using K clusters, relative to the assignment of K clusters in a randomized null distribution. Assuming K clusters, the gap statistic is given as:

$$\delta(K) = \text{mean}_M\{\log(W_{null(K)})\} - \log(W)$$

6) Obtain successive values $\delta(K+1)$, $\delta(K+2)$, $\delta(K+3)$, etc. This is done simply by incrementing the value for K and repeating the steps (1) through (5) above. Note that the optimal value of K (K_{optimal} , which is 3 in our demonstration case) is taken to be the first (i.e., lowest) K such that $\delta(K) \geq \delta(K+1) - s_{k+1}$:

$$K_{\text{optimal}} = \{K \mid \delta(K) \geq \delta(K+1) - s_{k+1}\}$$

Once the optimal K -value was determined for each MSA, we confirmed that these values accurately reflect the number of clusters by manually studying dozens of MSAs, including several MSAs corresponding of proteins known to constitute distinct conformations. We also examined several negative controls, such as CAP, an allosteric protein that does not undergo conformational change (19).

In manually annotating the alignments, we identified a vast array well-studied canonical allosteric domains and proteins. There may be many factors driving conformational change, and those cases for which the change is induced by the binding to a simple ligand (i.e., a simple consideration of *apo* or *holo* states) constitute only a very small subset of the conformational shifts observed in the PDB (Fig. S11). The gap statistic performed well in discriminating crystal structures that constitute such a diverse set, and this method has been validated using both domains and protein chains.

RMSD values were used to generate dendrograms for each of the selected MSAs. The dendrograms are constructed using the hierarchical clustering algorithm built into R, `hclust` (51), with UPGMA (mean values) used as the chosen agglomeration method (52).

Each domain is assigned to its respective cluster using the assigned optimal K-values as input to Lloyd's algorithm. For each sequence group, we perform 1000 K-means clustering simulations on the MDS coordinates, and take the most common partition generated in these simulations to assign each structure to its respective cluster.

We then select a representative structure from each of the assigned clusters. The representative member for each cluster is the member with the lowest Euclidean distance to the cluster mean, using the coordinates obtained by multidimensional scaling (see description above). These cluster representatives are then taken as the distinct conformations for this protein, and they are used for the binding leverage calculations and networks analyses (below).

Modeling Protein Motions by Directly Using Displacement Vectors from Alternative Conformations

As discussed, conformational changes may be modeled using vectors connecting pairs of corresponding residues in crystal structures of alternative conformations (which we term "ACT", for "absolute conformational transitions"). This more direct model of conformational change is especially straightforward to apply to single-chain proteins; applying this method on a database scale to multi-chain complexes introduces confounding factors related to chain-chain correspondence between such complexes when each complex has multiple copies of a given chain.

When we use ACT to apply the modified binding leverage framework for these proteins, we observe that our surface-critical residues are significantly more conserved than are non-critical residues (Fig. S17, left), and the same trend is observed when this is applied in our dynamical network analysis for identifying interior-critical residues (Fig. S17, right). There are too few human single-chain proteins to perform a reliable analysis in which conservation is evaluated using 1000 Genomes or ExAC data – for instance, only 9 (16) structures are such that 1000 Genomes (ExAC) SNVs overlap with interior-critical residues.

Evaluating Conservation of Critical Residues Using Various Metrics and Sources of Data

Conservation Across Species

All cross-species conservation scores represent the ConSurf scores, as downloaded from the ConSurf Server (53-56), in which scores for each protein chain are normalized to 0. Low (i.e., negative) ConSurf scores represent a stronger degree of conservation, and high (i.e., positive) scores designate weaker conservation. We perform cross-species conservation analysis on those proteins for which ConSurf files are available from the ConSurf server, and all ConSurf scores were calculated using default parameters, listed here:

- Homolog search algorithm: CSI-BLAST
- Number of iterations: 3
- E-value cutoff: 0.0001
- Proteins database: UniRef-90
- Maximum homologs to collect: 150
- Maximal %ID between sequences: 95
- Minimal %ID for homologs: 35
- Alignment method: MAFT-L-INS-i
- Calculation method: Bayesian
- Calculation method: JTT

Each individual point within the cross-species conservation plots (e.g., Figs. 3B and 3F, and Fig. S17) represents data from one structure: the value of the point for any given structure represents the mean conservation score for all residues within one of two classes: the set of N critical residues within a protein structure (surface or interior) or a randomly-selected set of N non-critical residues (with the same “degree”, see below)

within the same structure. The randomly selected non-critical set of residues was chosen in a way such that, for each critical residue with degree k (k being the number of non-adjacent residues with which the critical residue is in contact, see below), a randomly selected non-critical residue with the same degree k was included in the set. The distributions of non-critical residues shown are very much representative of the distributions observed when re-building the random set many times.

Note that the degree (i.e., k) of residue j is defined as the number of residues which interact with residue j , where residues adjacent to residue j in sequence are not considered, and an interaction is defined whenever any heavy atom in an interacting residue is within 4.5 Angstroms of any heavy atom in the residue j . We use degree as a measure of residue burial for several reasons. This metric for burial is consistent with our networks-based analysis for identifying interior-critical residues, as well as our use of residue-residue contacts in building networks for producing the ANMs. In addition, degree is also an attractive metric because it is discrete in nature, thereby allowing us to generate null distributions of non-critical residues with the exact same degree distribution.

Measures of Conservation Amongst Humans from Next-Generation Sequencing

All SNVs hitting protein-coding regions that result in amino acids changes (i.e., nonsynonymous SNVs) were collected from the phase 3 release of The 1000 Genomes Project (28). VCF files containing the annotated variants were generated using VAT (57). For nonsynonymous SNVs, the VCF files included the residue ID of the affected residue, as well as additional information (such as the corresponding allele frequency, the ancestral allele, and the residue type). To map the 1000 Genomes SNVs on to protein structures, FASTA files corresponding to the translated chain(s) of the respective transcript ID(s) were obtained using BioMart (58). FASTA files for each of the PDB structures associated with these transcript IDs (the PDB ID-transcript ID correspondence was also obtained using BioMart) were generated based on the ATOM records of the PDB files. For each given protein chain, BLAST was used to align the FASTA file obtained from BioMart with that generated from the PDB structure. The residue-residue correspondence obtained from these alignments was then used in order to map each SNV

to specific residues within the PDB. As a quality assurance mechanism, we confirmed that the residue type reported in the VCF file matched that specified in the PDB file.

ExAC SNVs were downloaded from the ExAC Browser (Beta), as hosted at the Broad Institute (29). SNVs were mapped to all PDBs following the same protocol as that used to map 1000G SNVs, and only non-synonymous SNVs in ExAC were analyzed. When evaluating SNVs from the ExAC dataset, minor allele frequencies (MAF) were used instead of DAF values. The ancestral allele is not provided in the ExAC dataset – thus, analysis is performed for MAF rather than DAF. However, we note that little difference was observed when using AF or DAF values with 1000 Genomes data, and we believe that the results with MAF values would generally be the same as those with DAF values. We also highlight the attractive feature of recapitulating the general conservation trends observed using a separate matrix.

When analyzing both 1000 Genomes and ExAC data, we consider only those structures in which at least one critical and one non-critical residue are hit by a non-synonymous SNV. This enables a more direct comparison between critical and non-critical residues, as comparisons between two different proteins would rely on the assumption of equal degrees of selection between such proteins.

Each individual point within the intra-human conservation plots (e.g., Figs. 3C, 3D, 3G, and 3H) represents data from one structure: the value of the point for any given structure represents the mean score (DAF or MAF, for 1000 Genomes or ExAC SNVs, respectively) for all critical (red bars) or non-critical (blue bars) residues to be hit by SNVs.

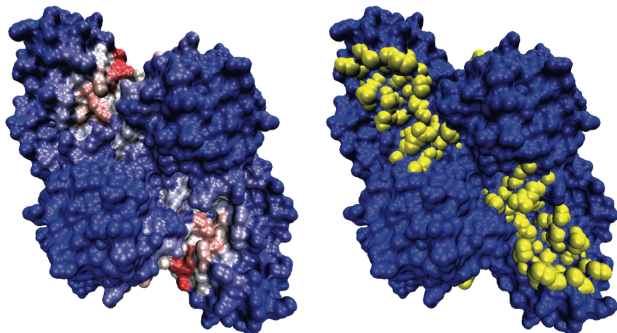
The *fraction* of rare SNVs to hit a particular “protein annotation” (described below) is defined to be the ratio of the number of rare non-synonymous SNVs in that annotation to the total number of non-synonymous SNVs to hit that annotation. An annotation for a given protein is simply the set of residues within a particular category, such as the set of all surface-critical residues (or alternatively the set of all interior-critical residues, or the set of non-critical residues). We define the term “rare” to mean that a 1000 Genomes SNV has a DAF value below a certain threshold – for instance, variable thresholds ranging from DAF = 0.05% to 0.50% are evaluated in Fig. S15. An SNV in

ExAC is defined to be rare if it has a MAF value below a certain threshold – variable thresholds ranging from MAF = 0.05% to 0.50% are evaluated in Fig. S16.

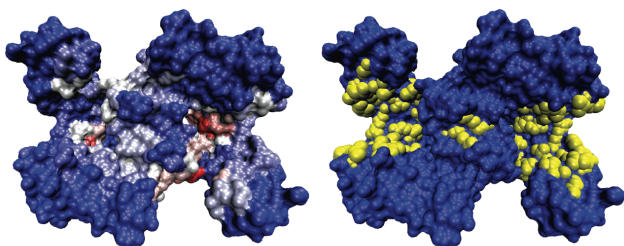
If a particular annotation, such as the set of surface-critical residues, has a rare SNV, then this rarity may potentially be a consequence of purifying selection acting to remove a deleterious SNV from the population pool (thereby making it rare). Such an annotation may thus be sensitive to sequence changes, and would thus be conserved. If there is a high fraction of such rare SNVs within the annotation, it provides further confidence to the claim that the annotation is conserved. Thus, a high fraction of rare SNVs is used as a signature for stronger conservation. Supporting this intuition, previous studies have observed that conserved genomic regions within the human population harbor higher fractions of rare SNVs (28, 30, 59).

Fig. S1: Canonical proteins with surface-critical and known ligand-binding sites

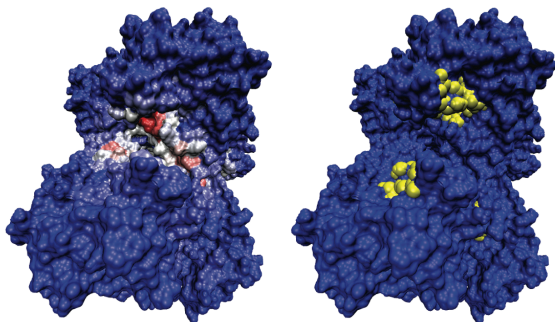
Each left image shows sites that are scored highly (i.e., surface-critical residues, in red), and each right image shows the residues (yellow) that actually come into contact of known ligands, based on the corresponding *holo* structure (Supp. Table 1).



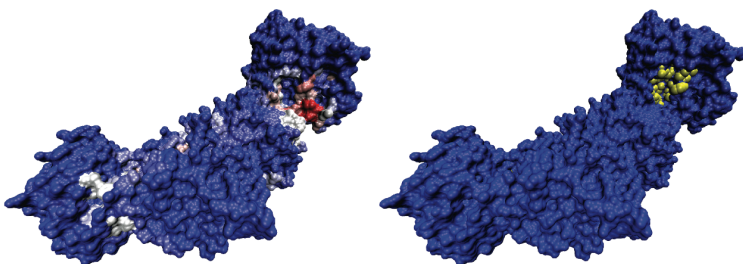
Phosphofructokinase (PDB ID 3pfk)



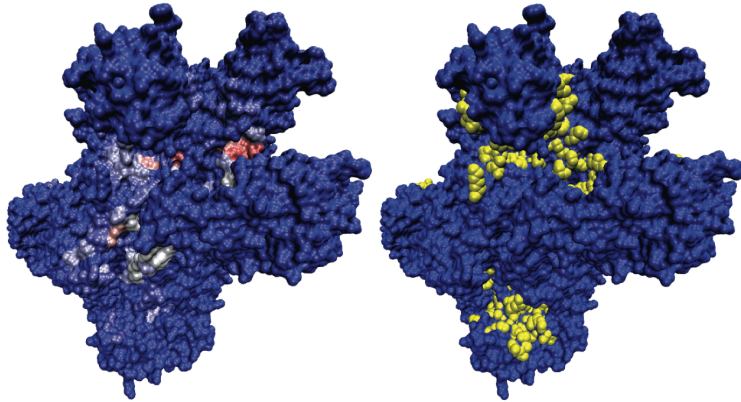
Adenylate Kinase (PDB ID 4ake)



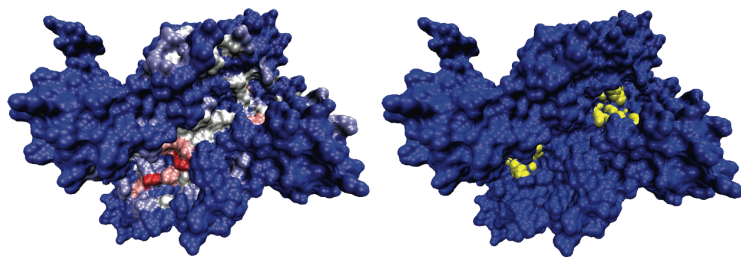
G6P-Deaminase (PDB ID 1cd5)



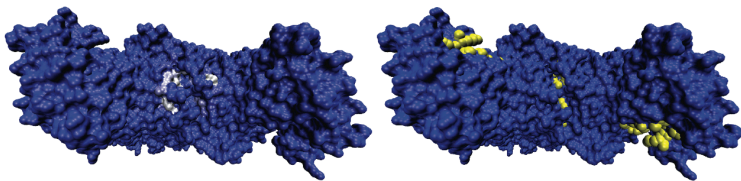
Trp Synthase (PDB ID 1bks)



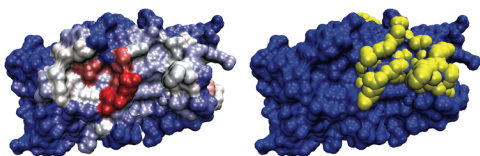
Glu Dehydrogenase (PDB ID 1nr7)



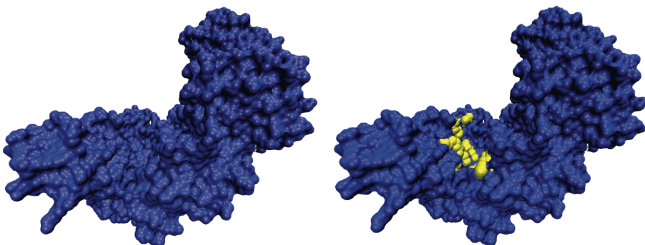
Thr Synthase (PDB ID 1e5x)



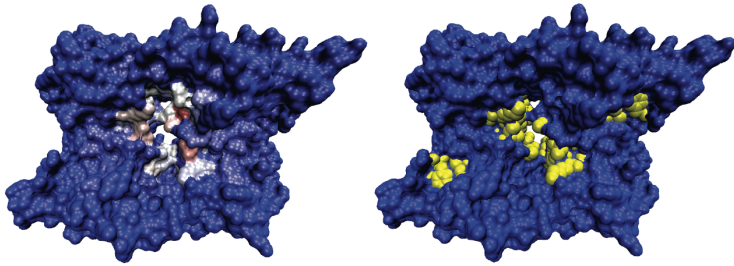
Malic Enzyme (PDB ID 1efk)



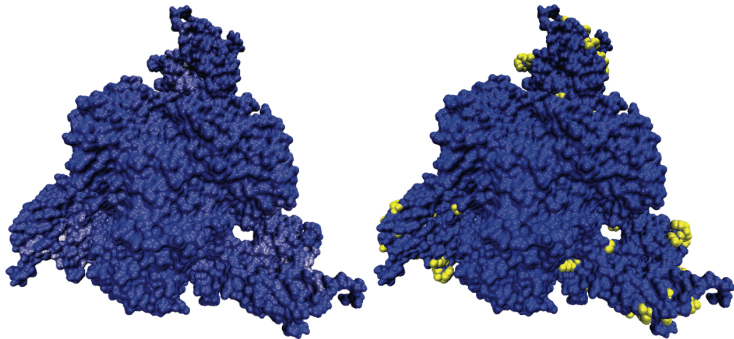
Tyr Phosphatase (PDB ID 2hnp)



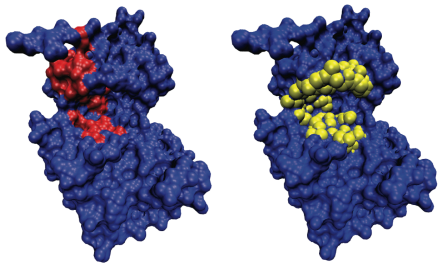
Arg Kinase (PDB ID 3ju5)



Phosphoribosyltransferase (PDB ID 1xtt)



Asp Transcarbamoylase (PDB ID 3d7s)



cAMP-dependent Kinase (PDB ID 1j3h)

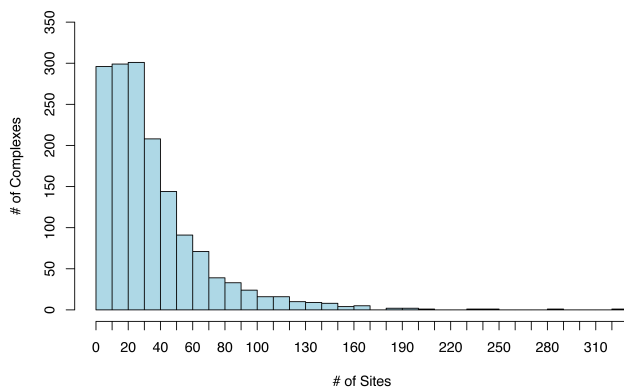


Fig. S2: Number of surface-critical sites per complex without thresholding

Complexes are taken from the the PDB biological assembly files. Shown is the distribution of the number of sites per complex. Without applying thresholds to the list of ranked surface-critical sites, the protein is often covered with an excess of identified critical sites.

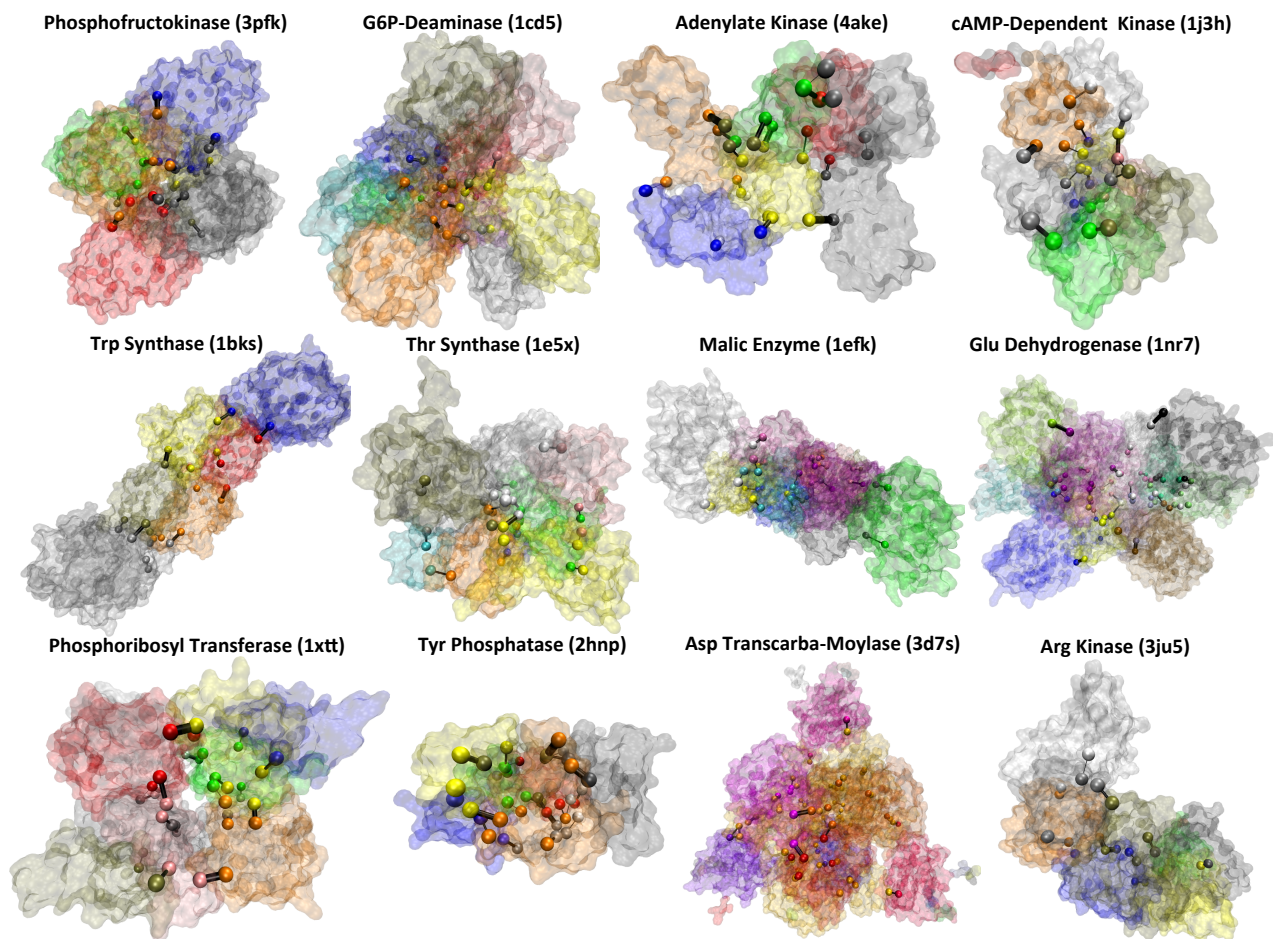


Fig. S3: Community partitioning for canonical systems

Different network communities are colored differently, and communities were identified using the dynamical network-based analysis with the GN formalism discussed in the main text and SI Methods. Residues shown as spheres are interior-critical residues, and they are colored based on community membership, and black lines connecting pairs of critical residues represent the highest-betweenness edges between the corresponding communities.

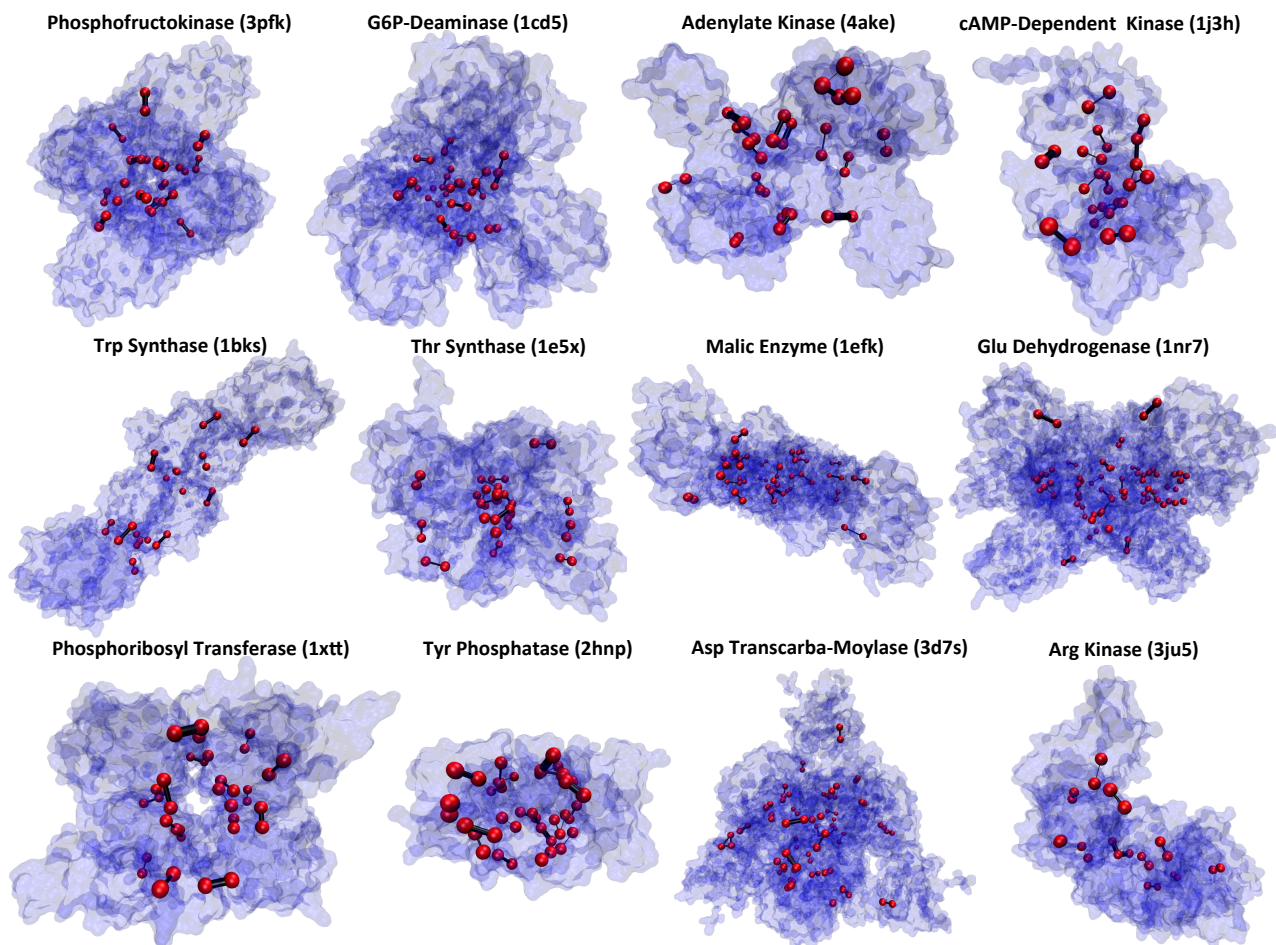


Fig. S4: Interior-critical residues highlighted in canonical systems

Shown above are the same proteins shown in Fig. S3, but with interior-critical residues highlighted in red spheres.

Fig. S5: Home page of the STRESS server (stress.molmovdb.org)

The server enables users to either provide PDB IDs or to upload their own PDB files for proteins of interest. Users may opt to identify surface-critical residues, interior-critical residues, or both.

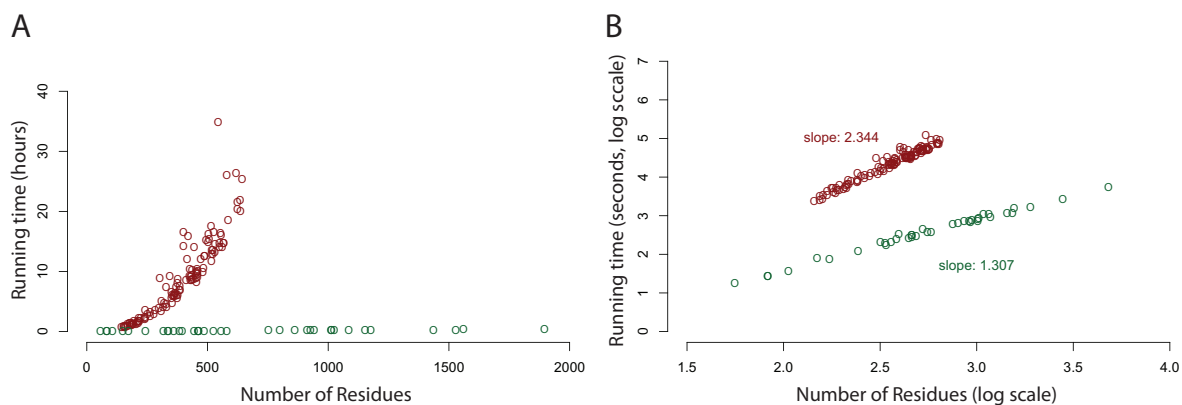


Fig. S6: Running times and optimization in the search of surface-critical residues

(A) Running times are shown for systems of various sizes. Shown in red are the running times without optimizing for speed. Performing local searching supported with hashing and implementing additional algorithmic optimizations for computational efficiency reduce running times considerably (in green), relative to a more naïve approach without optimization (in red). (B) The same data is represented as a log-log plot. The slopes of these two approaches demonstrate that our algorithm reduces the computational complexity by an order of magnitude. Our speed-optimized algorithm scales at $O(n^{1.3})$, where n is the number of residues.

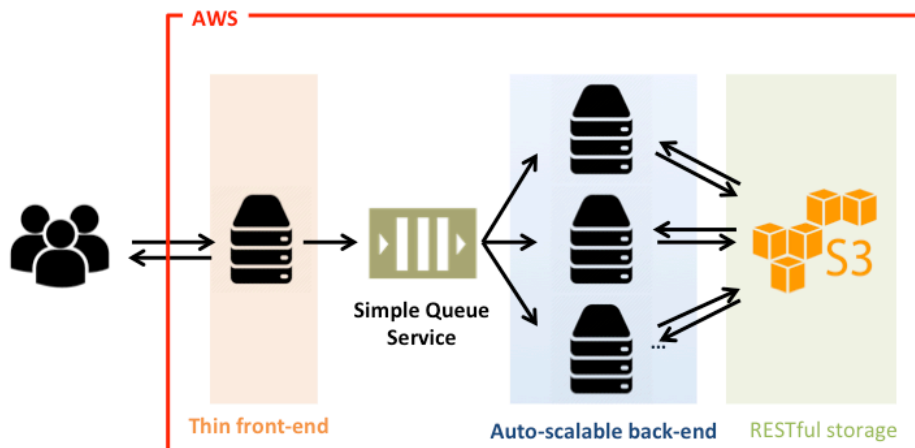


Fig. S7: Architecture of the STRESS server

A thin front-end server handles incoming user requests, and more powerful back-end servers perform the heavier algorithmic calculations. The back-end servers are dynamically scalable, making them capable of handling wide fluctuations in user demand. Amazon's Simple Queue Service is used to coordinate between user requests at the front end and the back-end compute nodes: when the front-end server receives a request, it adds the job to the queue, and back-end servers pull that job from the queue when ready. Source code is available through Github (github.com/gersteinlab/STRESS).

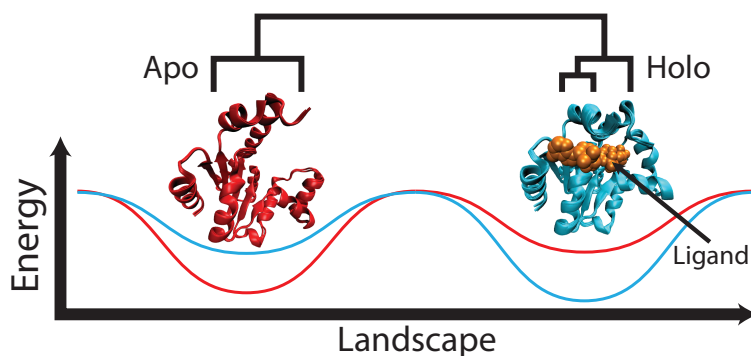


Fig. S8: Energy landscapes to describe distributions of different conformations

Energy landscape theory may be used to describe the relative populations of alternative biological states and conformations (for instance, active/inactive, or holo/apo). In the apo state, the landscape may take the form of the red curve, resulting in most proteins favoring the conformation shown in red. Once binding to ligand, the landscape becomes reconfigured to take the shape in the cyan curve, thereby shifting the distribution of conformations to that shown in cyan. One may use multiple structure alignments for domains or proteins to identify these distinct biological states in a database of structures. The schematized dendrogram represents the partitioning of these structures by a metric such as RMSD. The example shown is a multiple structure alignment of adenylate kinase. SCOP IDs of the *apo* domains: d4akea1 and d4akeb1; those of the *holo* domains: d3hpqb1, d3hpqa1, d2eckb1, d2ecka1, d1akeb1, and d1akea1.

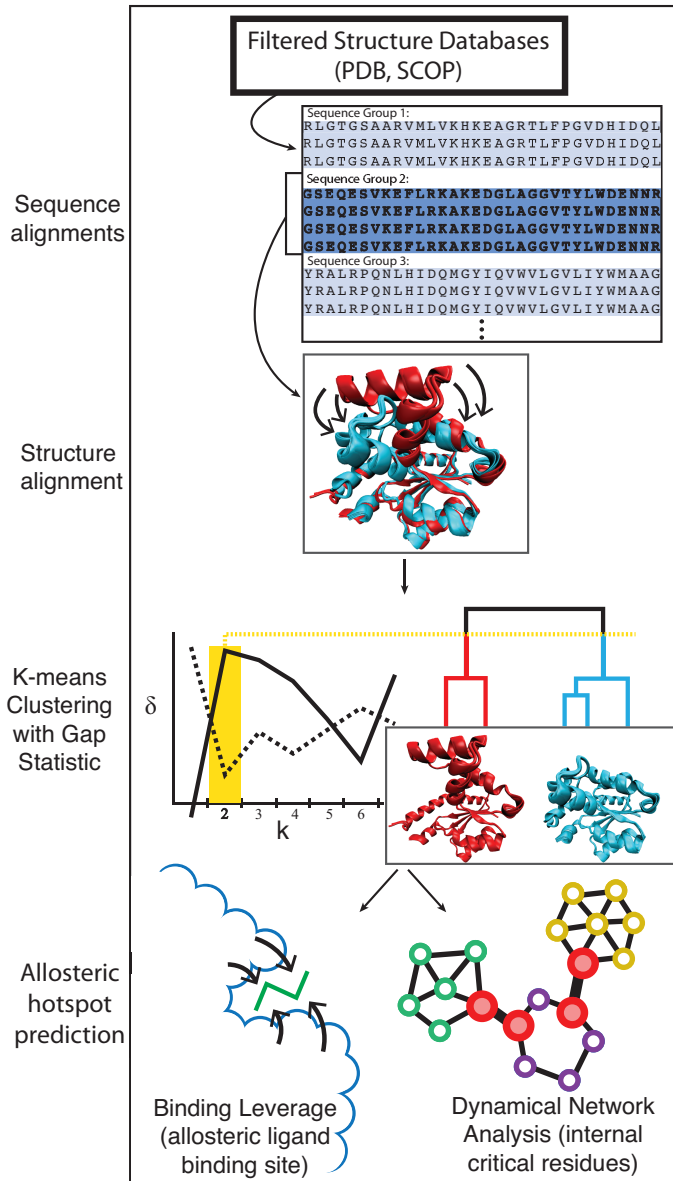


Fig. S9: Pipeline for identifying distinct conformations and critical residues

Top to bottom: BLASTClust is applied to the sequences corresponding to a filtered set of structures, thereby providing a large number of sequence-identical sets of proteins (i.e., “sequence groups”). For each sequence-identical group, a multiple structure alignment is performed using STAMP. The example shown here is adenylate kinase; details are provided in Fig. S8. Using the pairwise RMSD values in this structure alignment, the structures are clustered using the UPGMA algorithm, and K-means with the gap statistic (δ) is performed to identify the number of distinct conformations. The plot at left identifies 2 as the optimal value for K: the solid line represents $\delta(K)$ values at each value of K, and the dotted line represents $\delta(K+1) - s_{k+1}$ for each value of K (see SI Methods for details). The structures that exhibit multiple clusters (i.e., those with $K > 1$) are then taken to exhibit multiple conformations. Finally, surface-critical (bottom-left) and interior-critical (bottom-right) residues are identified on those proteins determined to exist as multiple conformations.

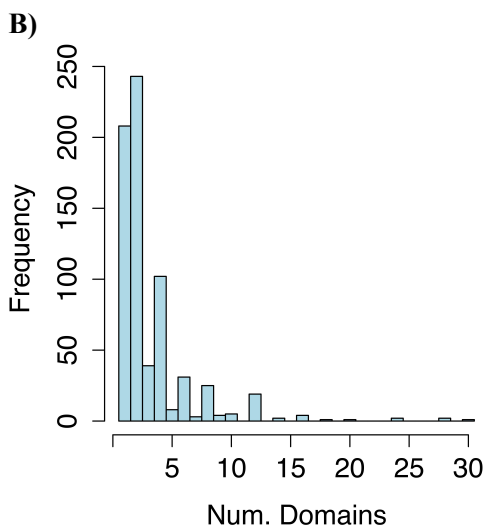
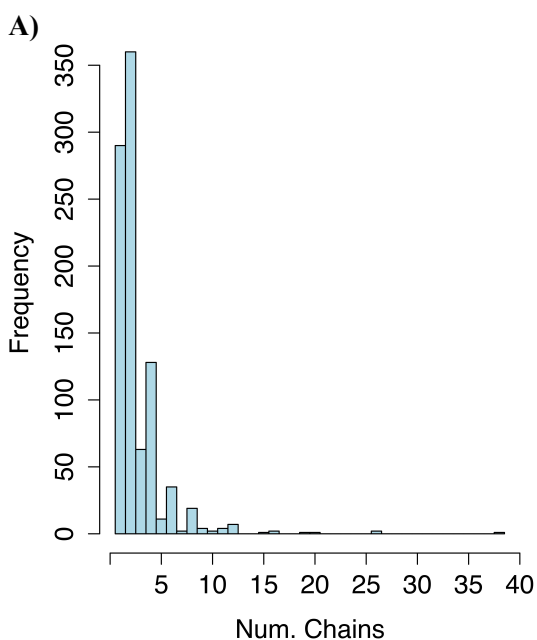
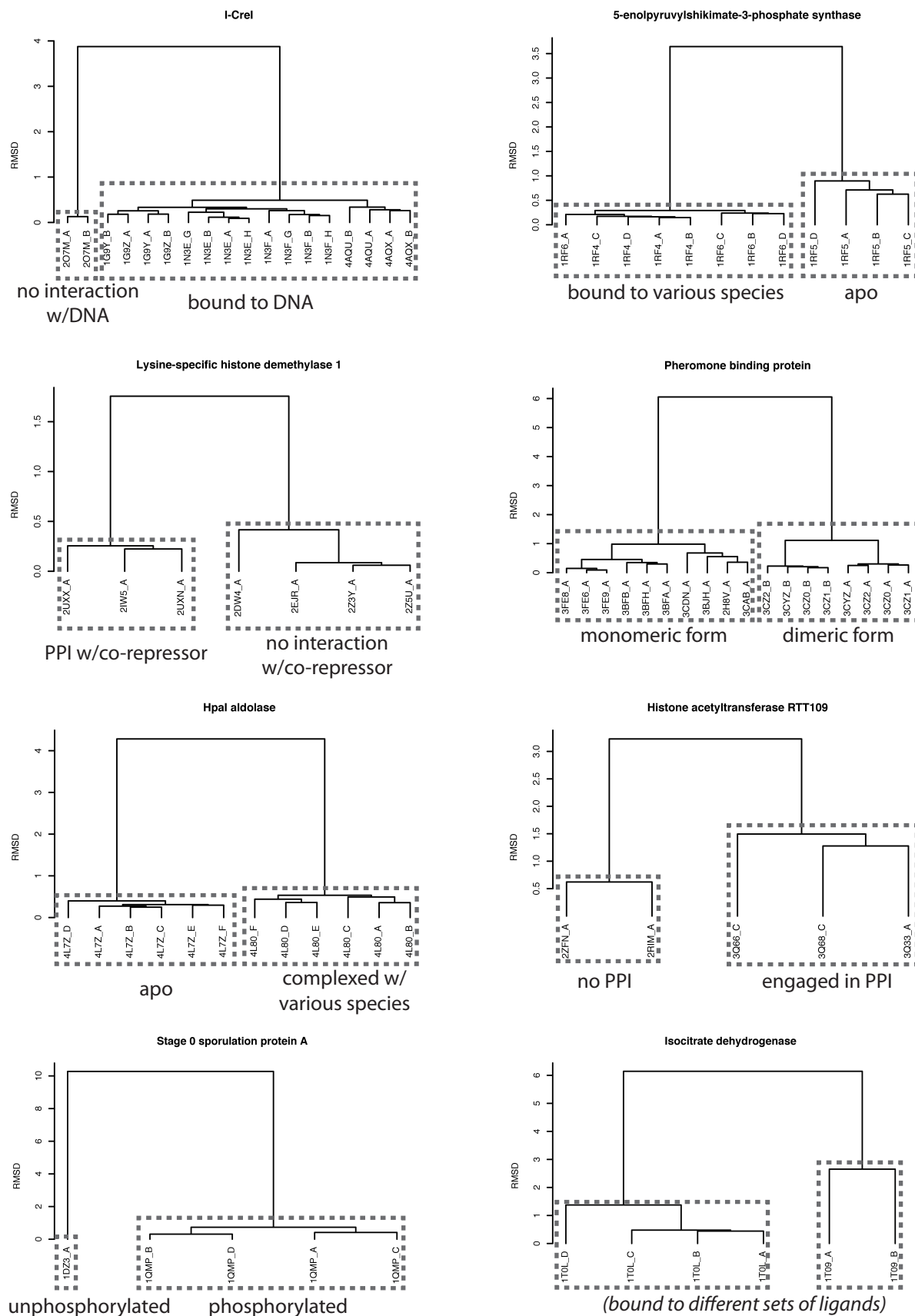
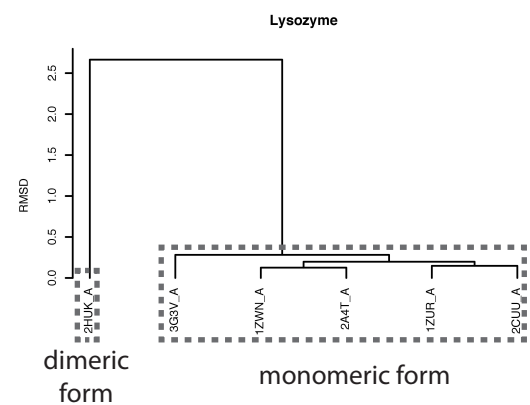
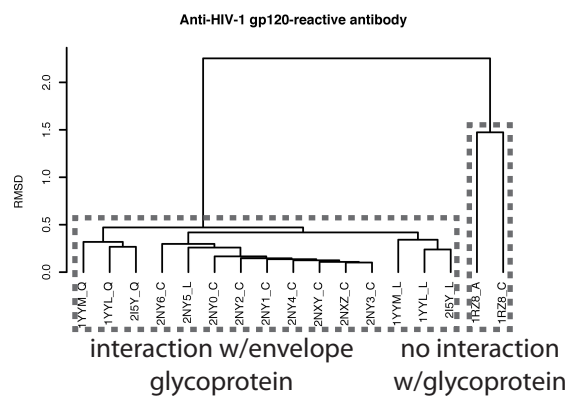
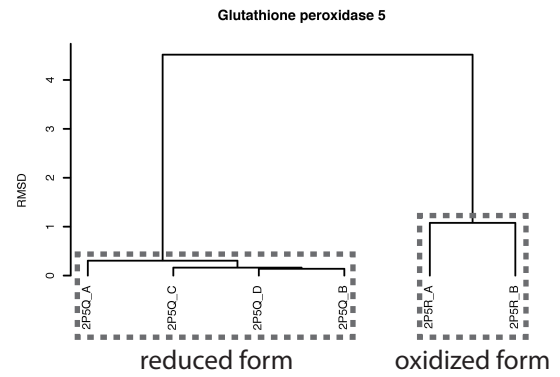
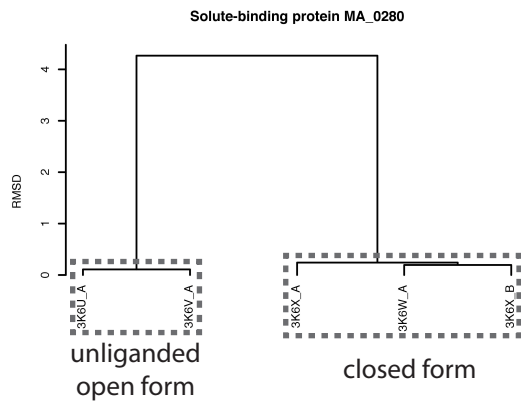
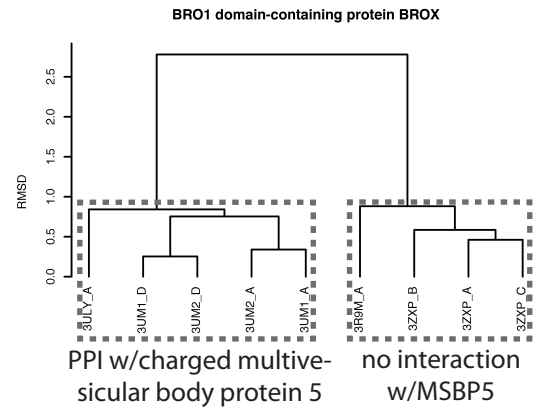
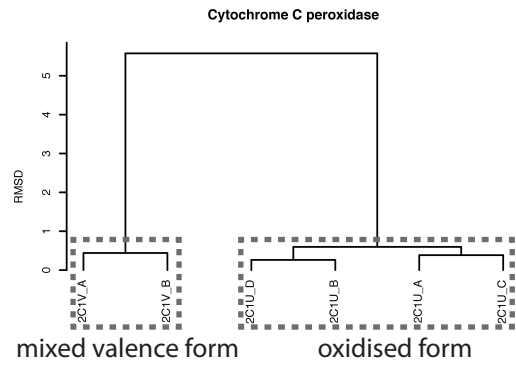
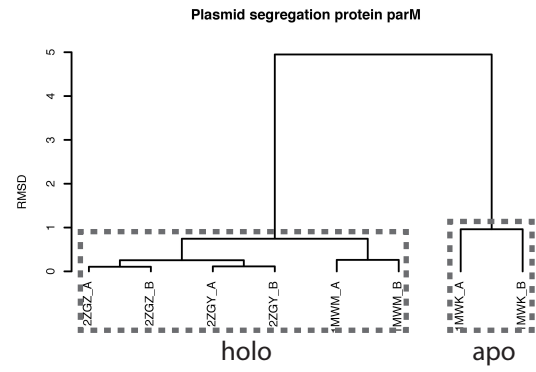
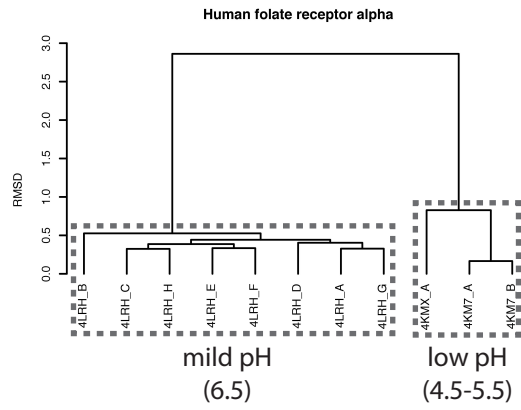


Fig. S10: Distributions of the number of chains and domains in the dataset of alternative conformations

Each structure the dataset of alternative conformations is taken from the first biological assembly file of the corresponding PDB. The structures in this database vary considerably in terms of size. Shown in panel (A) is the histogram representing the distribution for the number of chains in these biological assemblies, and shown in (B) is the corresponding distribution for the number of SCOP domains.

Fig. S11: Capturing alternative conformations in diverse biological contexts





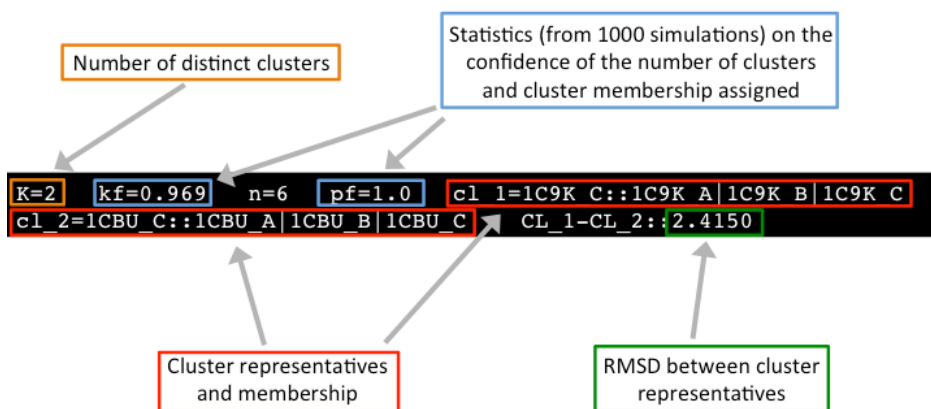
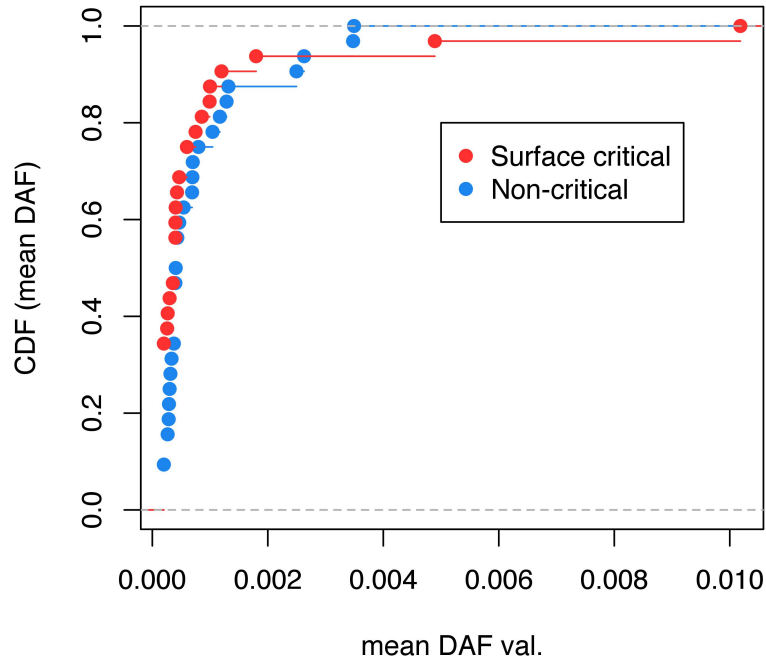
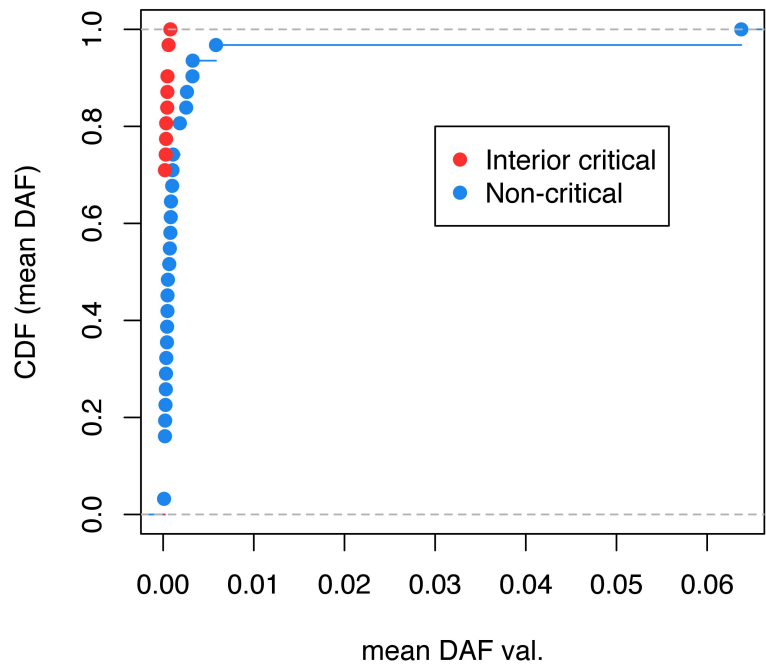


Fig. S12: A single annotated entry from our database of alternative conformations

The clustering for the protein adenosylcobinamide kinase is shown. Two distinct conformations are represented in the ensemble of structures. The measure kf designates the fraction of times that the optimal value of K (here, $K=2$) was obtained out of 1000 simulations in which the algorithm (K-means with the gap statistic) obtained this particular value of K . The high kf value (0.969) signifies that these structures are very well clustered into two groups. n designates the number of distinct structures (PDB chains in this case) in the multiple structure alignment. pf designates the fraction of times (out of 1000 simulations of running Lloyd's algorithm, the standard K-means algorithm) that this particular set of structure-group assignments were assigned. In this example, for all 1000 simulations, 1C9K_C and 1C9K_A were clustered in one group, and 1CBU_A, 1CBU_B, 1CBU_C clustered together. Within each cluster (the two clusters shown as two red boxes), the chain preceding the "::" tag designates the cluster representative (i.e., the structure closest to the Euclidean centroid of the cluster). The last field gives the RMSD values between cluster representatives. See the header information within Supp. File 1 for further details.

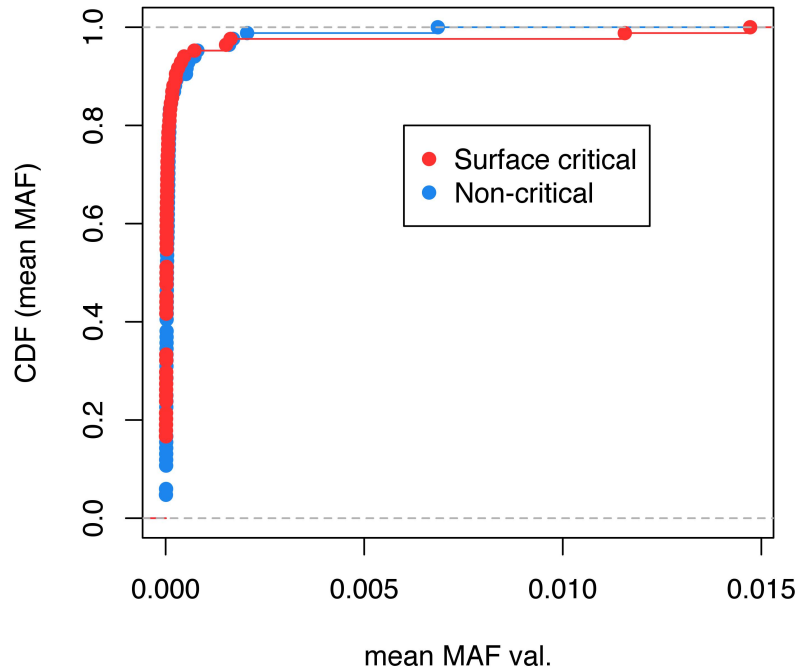


A) Cumulative distribution functions for mean DAF values of surface-critical and non-critical residues (p-val = 0.159, KS test)

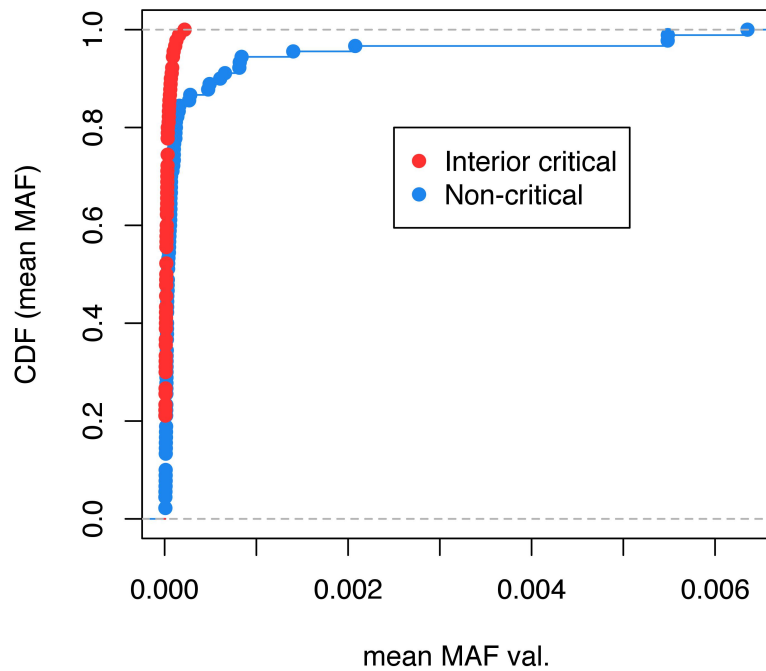


B) Cumulative distribution functions for mean DAF values of interior-critical and non-critical residues (p-val = 1.79e-4, KS test)

Fig. S13: Potential shifts in DAF distributions (in 1000 Genomes) using two-sample Kolmogorov-Smirnov tests



A) Cumulative distribution functions for mean minor allele frequencies of surface-critical and non-critical residues (p-val = 9.49e-2, KS test)



B) Cumulative distribution functions for mean minor allele frequencies of interior-critical and non-critical residues (p-val = 1.75e-4, KS test)

Fig. S14: Potential shifts in mean minor allele frequency distributions (in ExAC) using two-sample Kolmogorov-Smirnov tests

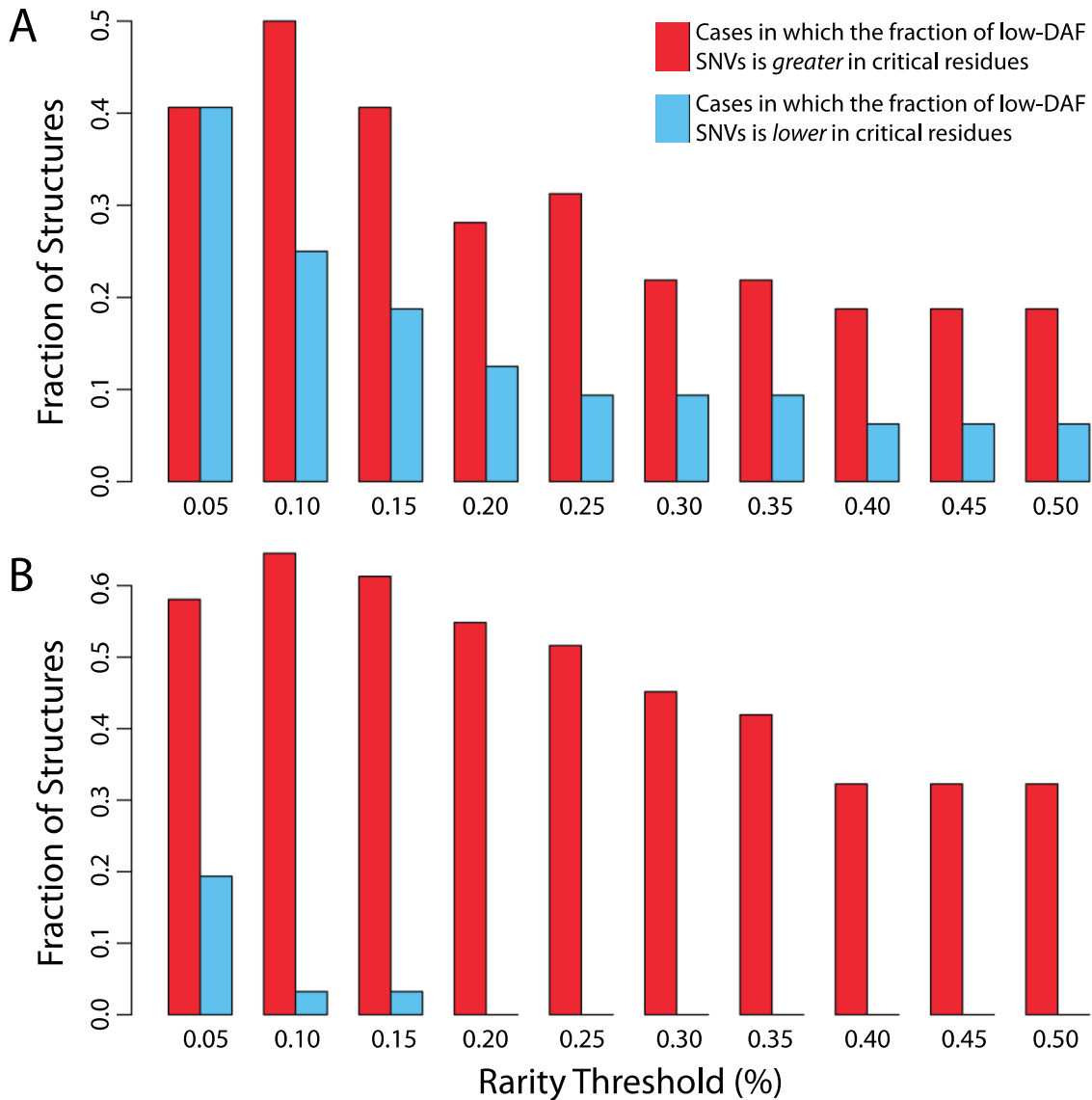


Fig. S15: Measuring relative conservation by the fraction of rare (low-DAF) variants using 1000 Genomes data

Protein regions with high fractions of *rare* variants are believed to be more sensitive to sequence variants than are other regions (thereby explaining why such variants occur infrequently in the population). Here, a rarely occurring SNV within the human population is defined to be one with a DAF less than or equal to the rarity thresholds given on the y-axis. We consider all structures such that at least one critical and at least one non-critical residue are hit by a 1000 Genomes non-synonymous SNV. Distributions in which the critical residues are defined to be the surface-critical (*A*) and interior-critical (*B*) residues are shown. For varying thresholds to define rarity, there are more structures in which the fraction of rare variants is higher in critical residues than in non-critical residues. Cases in which the fraction is equal in both categories are not shown. (*A*) represents data from 31 structures, and (*B*) represents data from 32 structures.

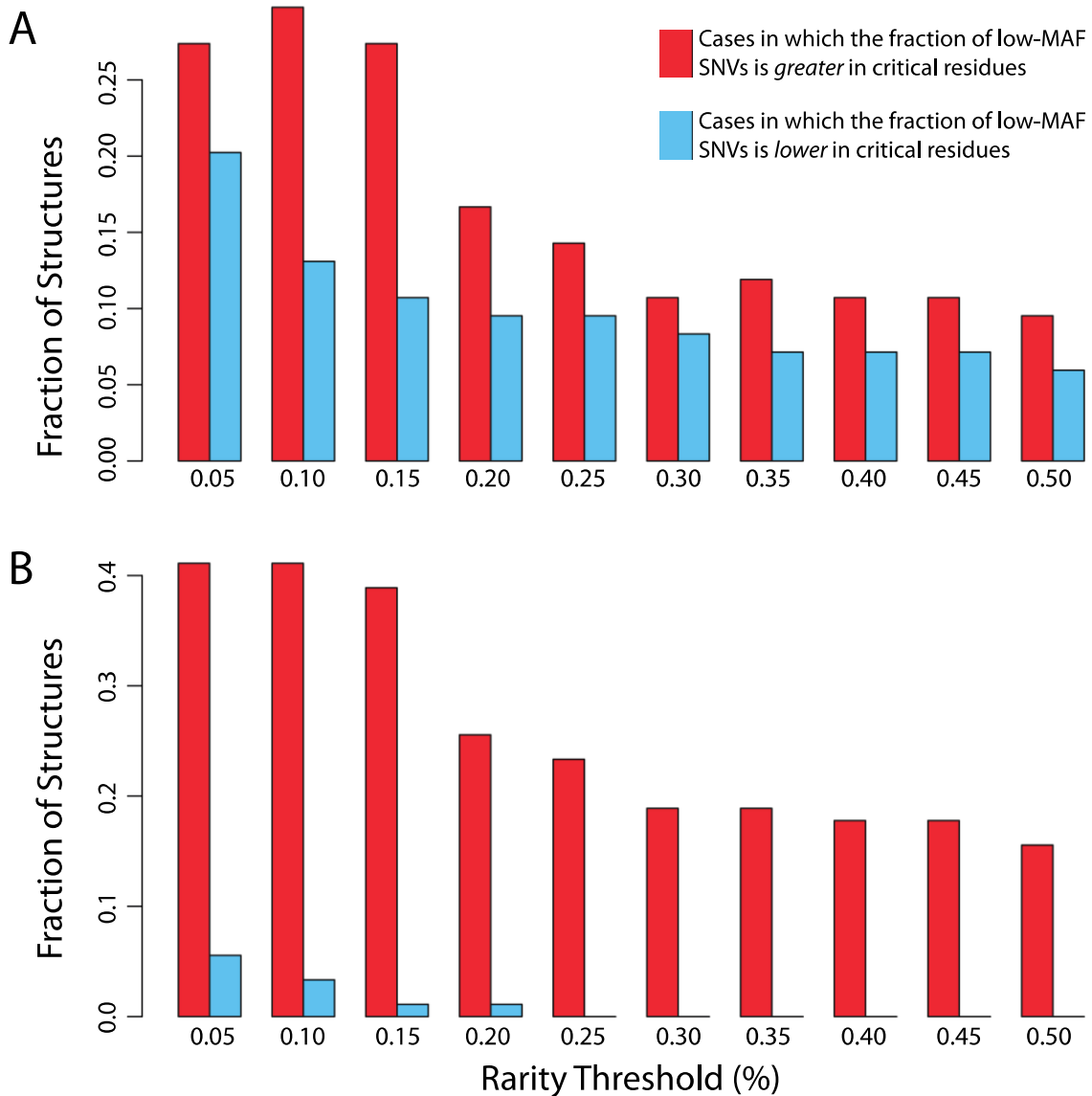


Fig. S16: Measuring relative conservation by the fraction of rare (low-MAF) variants using ExAC data

Protein regions with high fractions of *rare* variants are believed to be more sensitive to sequence variants than are other regions (thereby explaining why such variants occur infrequently in the population). Here, a rarely occurring SNV within the human population is defined to be one with a MAF less than or equal to the rarity thresholds given on the y-axis. We consider all structures such that at least one critical and at least one non-critical residue are hit by a non-synonymous SNV in the ExAC dataset. Distributions in which the critical residues are defined to be the surface-critical (*A*) and interior-critical (*B*) residues are shown. For varying thresholds to define rarity, there are more structures in which the fraction of rare variants is higher in critical residues than in non-critical residues. Cases in which the fraction is equal in both categories are not shown. (*A*) represents data from 90 structures, and (*B*) represents data from 84 structures.

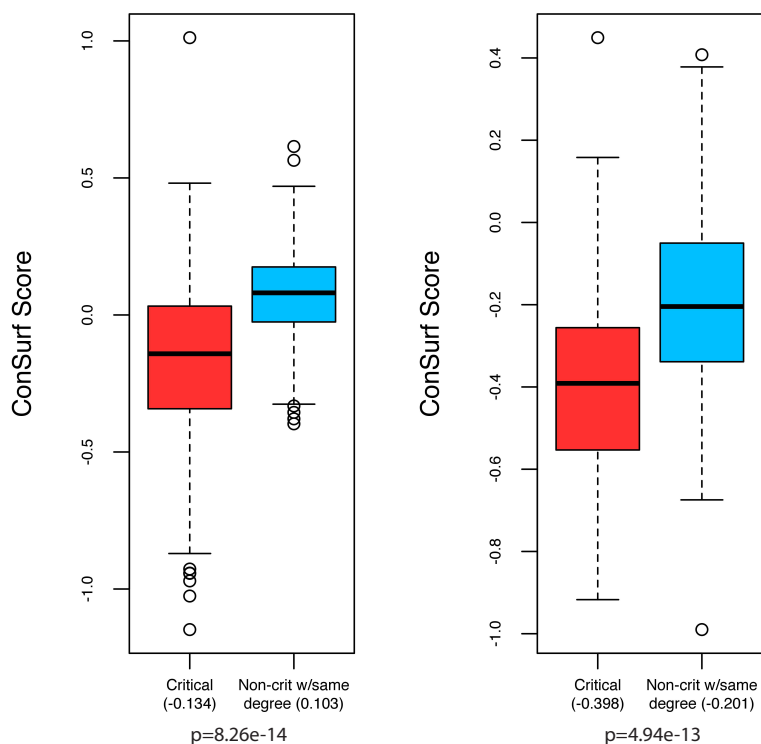


Fig. S17: Modeling protein conformational change through a direct use of crystal structures from alternative conformations using absolute conformational transitions (ACT)

Left: Distributions (155 structures) of the mean conservation scores on surface-critical (red) and non-critical residues with the same degree of burial (blue). *Right:* Distributions (159 structures) of the mean conservation scores for interior-critical (red) and non-critical residues with the same degree of burial (blue). Mean values are given in parentheses. Results for single-chain proteins are shown, and p-values were calculated using a Wilcoxon rank sum test.

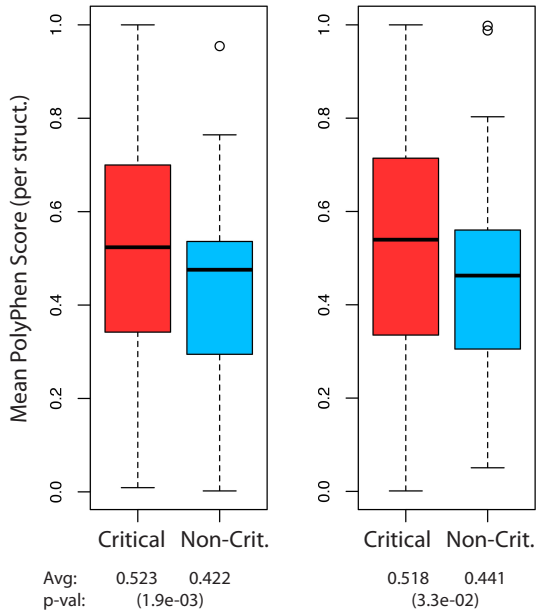


Fig. S18: Mean PolyPhen scores for critical- and non-critical residues, as identified by ExAC

Left: Distributions (64 structures) of mean PolyPhen values on surface-critical residues (red) and non-critical residues (blue). *Right:* Distributions (70 structures) of mean PolyPhen values on interior-critical residues (red) and non-critical residues (blue). Overall mean values and p-values are given below plots. Note that higher PolyPhen scores denote more damaging variants.

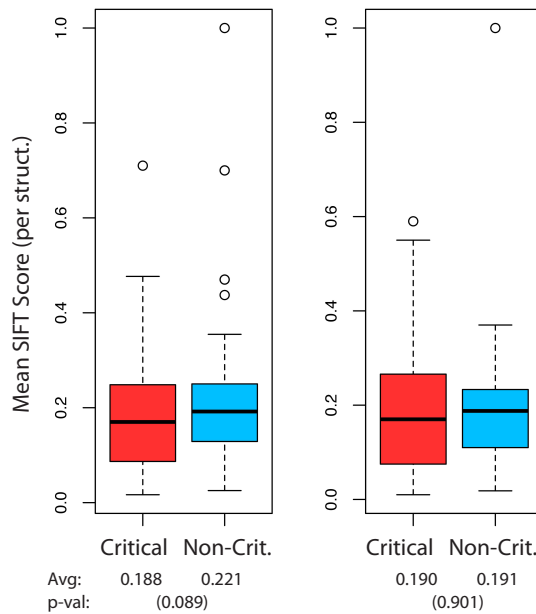


Fig. S19: Mean SIFT scores for critical- and non-critical residues, as identified by ExAC

Left: Distributions (63 structures) of mean SIFT values on surface-critical residues (red) and non-critical residues (blue). *Right:* Distributions (65 structures) of mean SIFT values on interior-critical residues (red) and non-critical residues (blue). Overall mean values and p-values are given below plots. Note that lower SIFT scores denote more damaging variants.

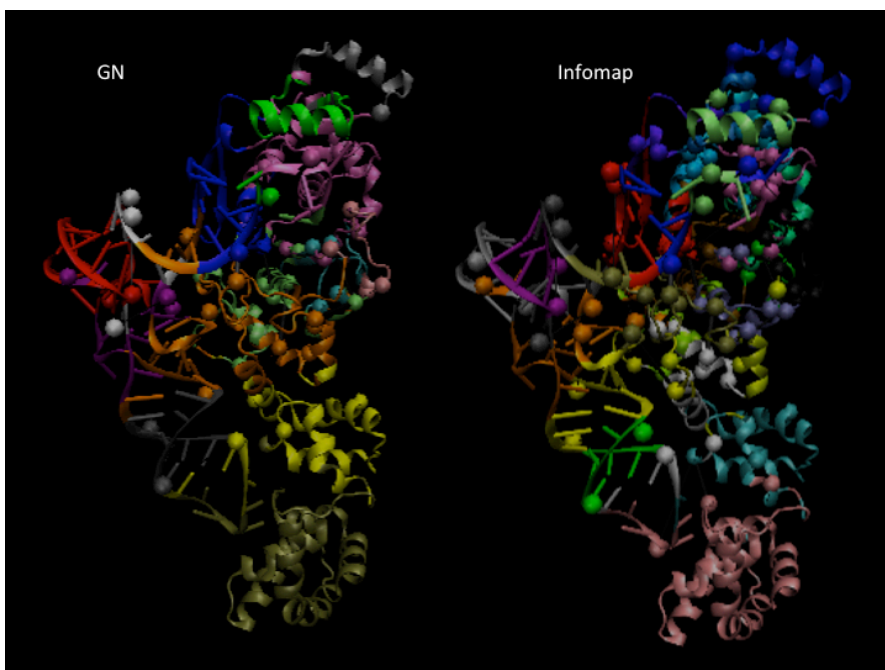


Fig. S20: Network modularization by GN and Infomap

Different colors correspond to different communities. Network modularization by the GN (left) and Infomap (right) algorithms are shown for the crystal structure of glutamyl-tRNA synthetase complexed with tRNA(Glu) and glutamol-AMP (PDB 1N78).

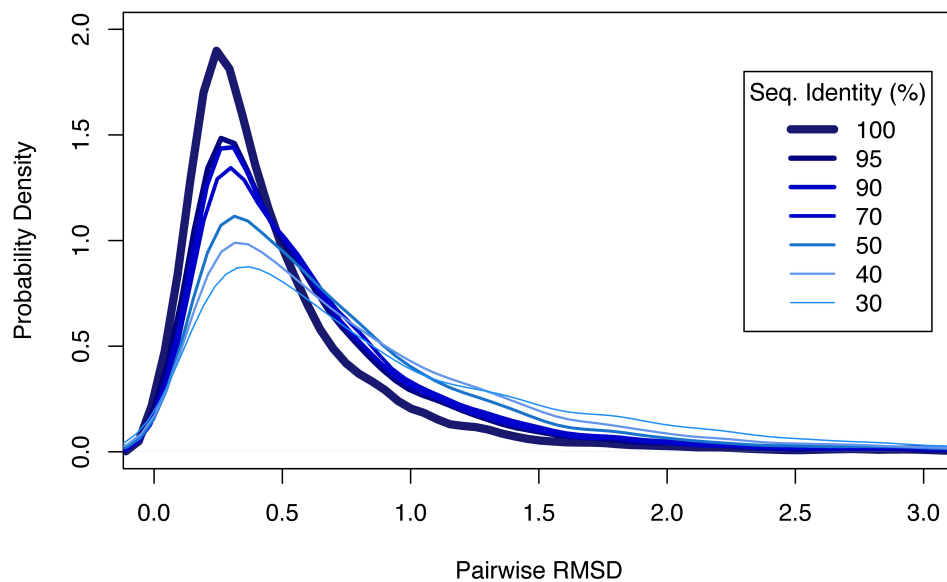


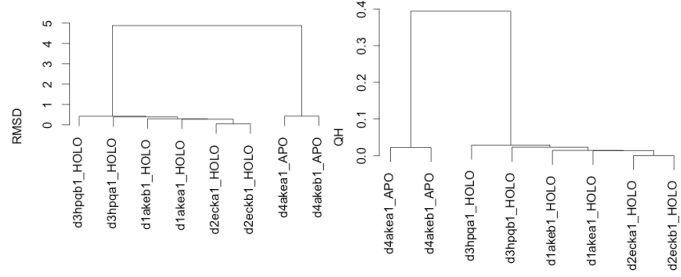
Fig. S21: Probability distributions of pairwise-RMSD by sequence identity

Distributions for average pairwise RMSD values across domains within all multiple structure alignments at varying levels of sequence identity.

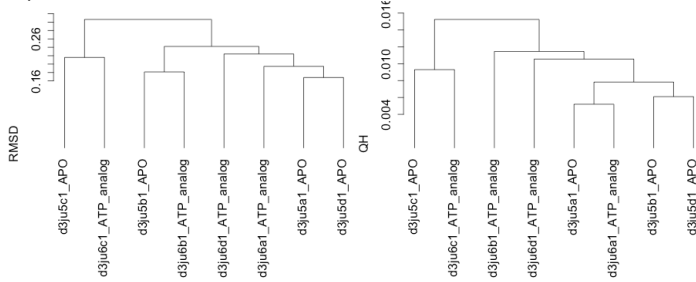
Fig. S22: Representative clustering of domains based on RMSD and Q_H : RMSD generally matches the clustering obtained when using Q_H

Shown are the dendrograms for domains in adenylate kinase (*A*), arginine kinase (*B*), calcyclin (*C*), and catabolite activator protein (*D*)

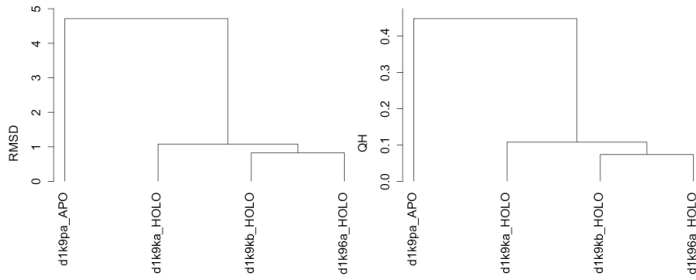
A)



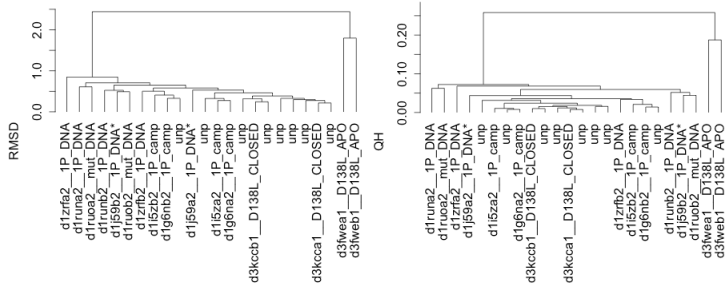
B)



C)



D)



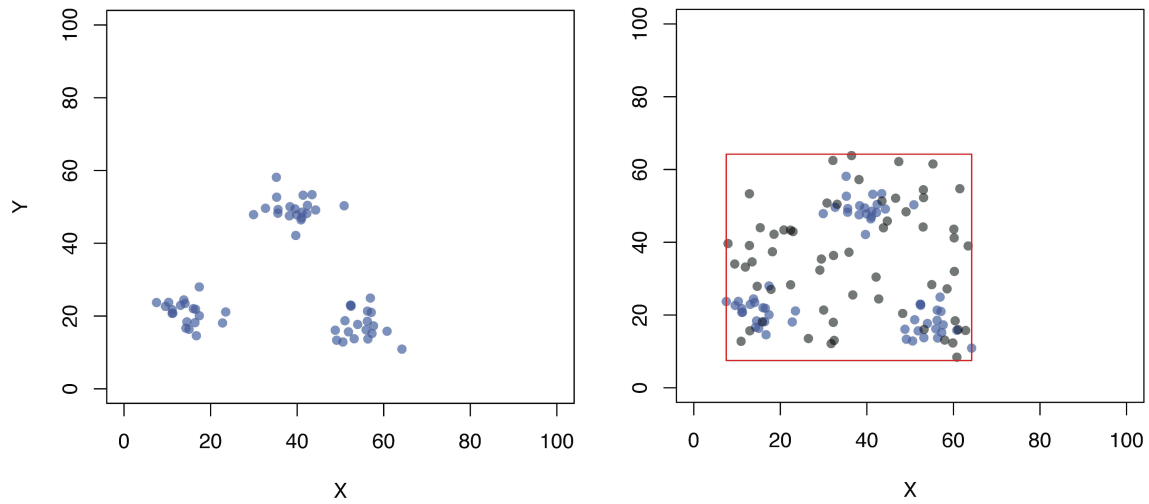


Fig. S23: Intuition behind the k-means algorithm with the gap statistic

The objective is to identify the ideal number of clusters to describe the observed data of 60 points (in blue). This entails defining how well-clustered our observed data appears (given an assigned number of clusters, K) relative to a null model consisting of a randomly distributed set of 60 points (grey) that fall within the same variable ranges as the observed data. Further details are provided by Tibshirani et al, 2001.

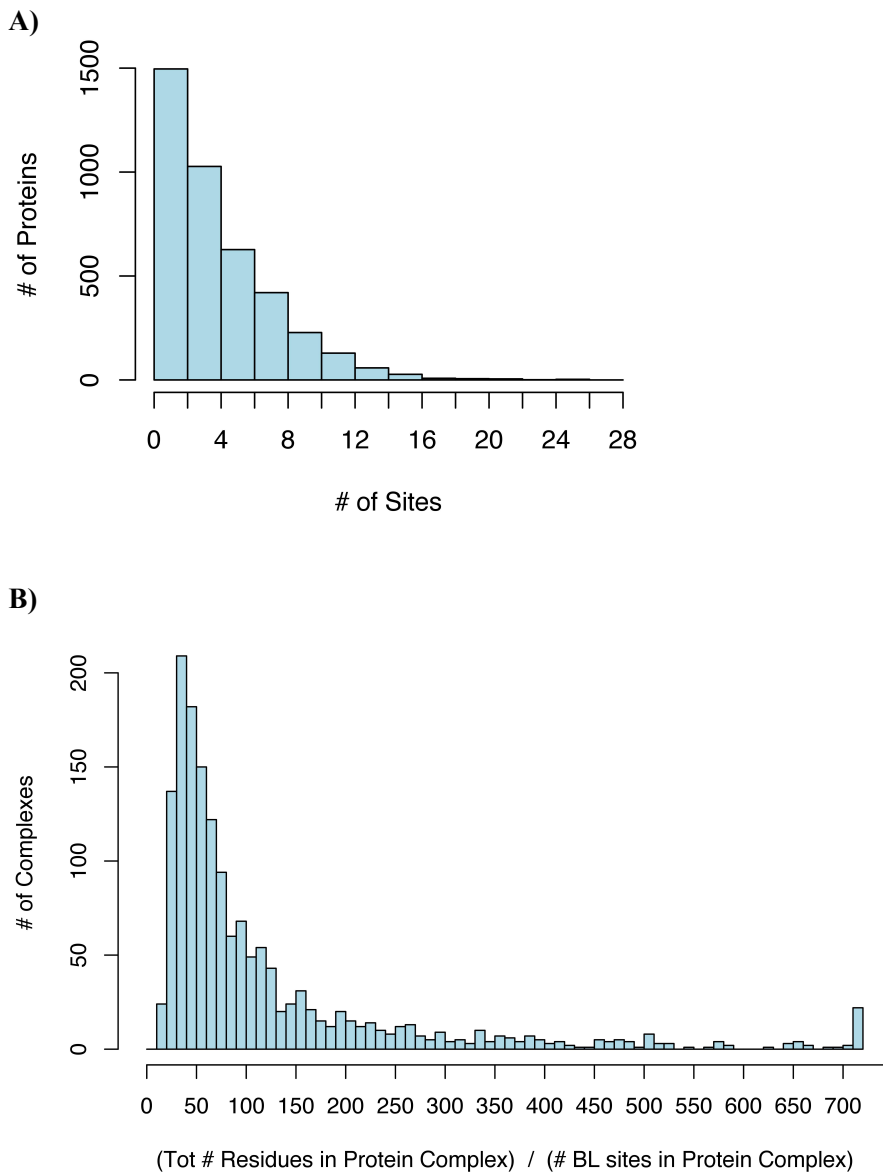


Fig. S24: Quantifying the number of distinct surface-critical sites

(A) The distribution of the number of surface-critical sites per PDB chain; (B) The density of surface-critical residues with respect to the total number of residues in the biological assembly (here referred to as a “complex”, though in some cases, the biological assembly may in fact be a single chain).

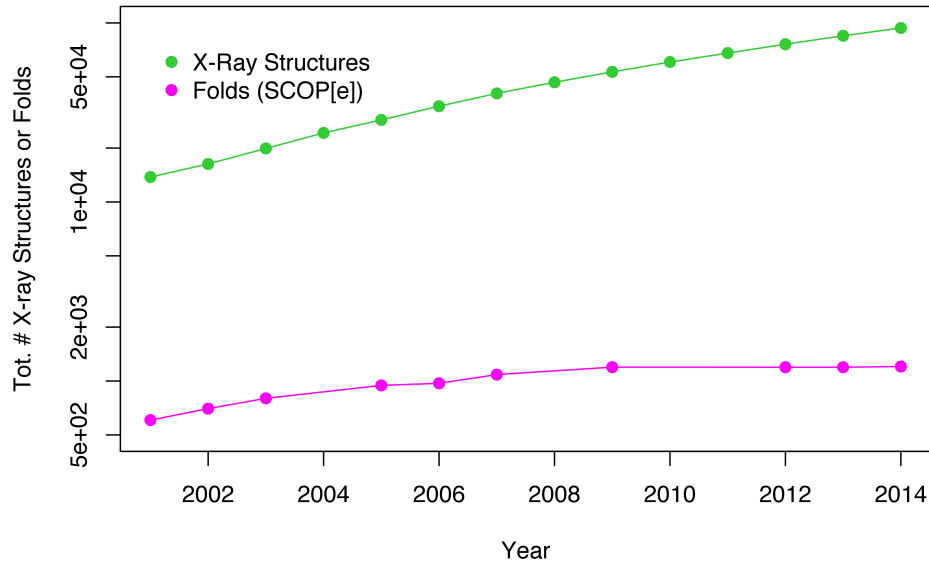


Fig. S25: Growth rate of deposited PDB structures, and the concomitant growth rate in the number of folds (as defined by CATH and SCOP)

The growing appreciation for dynamic behavior and the importance of conformational heterogeneity is being facilitated by a growing redundancy within the PDB. Such redundancy is represented, for instance, when the same protein is structurally resolved under different conditions, potentially resulting in alternative conformations.

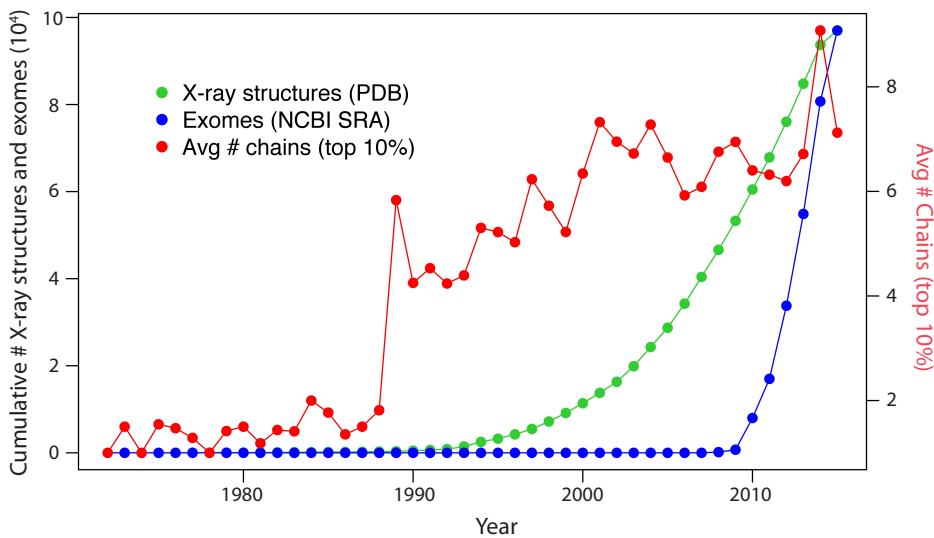


Fig. S26: Trends in data generation point to growing opportunities for leveraging sequence variants to study structure (and vice versa)

The volume of sequenced exomes is outpacing that of structures, while solved structures have become more complex in nature. Red: Average number of chains per PDB (considering the biological assembly PDB files for the top 10% of PDBs for a each year). Green: Cumulative number of X-Ray structures deposited in the PDB. Blue: Cumulative number of exomes stored in the NCBI Sequence Read Archive (SRA). All data were downloaded in May 2015.

HOLO	APO
1ake (AP5)	4ake
3cep (G3P, IDM, PLP)	1bks (PLP)
1hor (AGP, PO4 , [& 16G in pdb 1HOT])	1cd5
2c2b (SAM , [& LLP in pdb 2c2g])	1e5x
1gz3 (ATP, FUM, oxl)	1efk (MAK)
1atp (ATP)	1j3h
1hwz (GLU, GTP, NDP [& ADP in PDB 1NQT])	1nr7
1xtu (CTP, USP)	1xtt (<i>ACV</i> , USP)
1aax (BPM [& 892 in PDB 1T49])	2hnp
7at1 (ATP, MAL, PCT [& CTP in PDB 1RAC], [& PAL in PDB 1D09])	3d7s
3ju6 (ANP, ARG)	3ju5
6pfk (PGA [& F6P + ADP in PDB 4PFK])	3pfk (<i>PO4</i>)

Table S1: Set of 12 canonical proteins, organized by state (*apo* or *holo*)

Ligands are given in parentheses (those in bold text designate the ligand used to define residues involved in canonical ligand-binding interactions).

PDB	Fract Protein Hit	Fract Known Bio Sites Hit
3pfk	0.108	1
4ake	0.287	1
1cd5	0.09	0.5
1j3h	0.06	1
1bks *	0.132	0.25
1e5x	0.151	0.667
1efk	0.032	0
1nr7	0.071	0.75
1xtt	0.111	1
2hnp	0.672	1
3d7s *	0.052	0
3ju5	0.017	0
<i>Avg</i> s	0.15 (0.16 w/o 3d7s)	0.60 (0.65 w/o 3d7s)

Table S2: Identifying known ligand-binding sites

The 2nd column designates the fraction of residues that constitute surface-critical residues, and the 3rd column represents, for each structure, the fraction of known ligand-binding sites that strongly overlap with surface-critical sites.

PDB	Fract protein hit by our predictions	Fract protein actually occupied by biological ligand-binding sites
3pfk	0.11	0.13
4ake	0.29	0.17
1cd5	0.09	0.08
1j3h	0.06	0.08
1bks	0.13	0.07
1e5x	0.15	0.09
1efk	0.03	0.08
1nr7	0.07	0.16
1xtt	0.11	0.16
2hnp	0.67	0.11
3d7s *	0.05	0.07
3ju5	0.02	0.08

Table S3: Do surface-critical sites occupy an exceedingly large fraction of the protein?

For most proteins in the canonical set, the fraction of the protein occupied by surface-critical residues roughly matches the fraction of residues known to be directly involved in ligand binding. For most proteins (blue), the fraction of critical-surface residue is actually lower than that of known ligand-binding residues.

n	Fract Known Bio Sites Hit (w/ and w/o 3d7s)
6	0.60 (0.65)
5	0.62 (0.67)
4	0.69 (0.75)
3	0.74 (0.76)
2	0.81 (0.81)
1	0.86 (0.85)

Table S4: Capturing known-ligand binding sites at varying thresholds

Here, n designates the number of residues within a surface-critical site that overlap with known ligand-binding residues. For the calculations reported above and in the main text, this value is taken to be $n=6$ (because each surface-critical site typically has 10 residues, and never has more than 10 residues, this criterion enforces that a majority of surface-critical residues within a given site overlap with known ligand-binding residues in order to be counted as a site match). However, as this threshold is relaxed to lower n , the fraction of captured known ligand-binding sites improves rapidly, suggesting that surface-critical sites generally lie close to known ligand binding sites in many cases.

Protein (PDB, # residues)	Community Detection Method: GN InfoMap			
	Modularity	# Comm.	# Critical Residues	% of GN critical residues which match those in Infomap (expected)
tRNA synthetase (1N78, 542)	0.71 0.68	14 25	47 109	0.28 (0.20)
Adenylate kinase (4AKE, 428)	0.73 0.70	11 20	39 82	0.90 (0.19)
Arginine Kinase (3JU5, 728)	0.72 0.69	12 28	41 142	0.22 (0.19)
Tyrosine Phosphatase (2HNP, 278)	0.59 0.59	7 15	27 70	0.26 (0.25)
Phosphoribosyltransferase (1XTT, 846)	0.72 0.68	9 32	36 174	0.22 (0.21)
cAMP-dep. PK (1J3H, 332)	0.66 0.64	11 19	36 78	0.33 (0.23)
Anthranilate synthase (1I7Q, 1418)	0.75 0.69	12 46	51 288	0.31 (0.20)
Malic enzyme (1EFK, 2212)	0.81 0.72	17 70	74 425	0.18 (0.19)
Threonine synthase (1E5X, 884)	0.73 0.69	13 36	43 192	0.28 (0.22)
G-6-P Deaminase (1CD5, 1596)	0.79 0.72	18 54	58 266	0.16 (0.17)
Phosphofructokinase (3PFK, 1276)	0.76 0.68	10 51	45 307	0.24 (0.24)
Tryptophan synthase (1BKS, 1294)	0.77 0.69	10 46	41 284	0.24 (0.22)
Means	0.73 0.68	12.0 36.8	44.8 201.4	0.3

Table S5: Comparing the two network module identification algorithms GN & Infomap

Though both GN (values to the left of “|” symbols throughout the table) and Infomap (values to the right) decompose networks to give similar modularity, the number of communities, and hence the number of critical residues connecting communities, is substantially larger when decomposing networks using Infomap than using GN.

