

Supplemental Information

1 – Supplemental Figures

2 – Supplemental Tables

3 – Supplemental Methods

3.1 Identifying Potential Allosteric Residues

3.1-a Identifying Surface-Critical Residues

3.1-a-i Monte Carlo Simulations & Parameterization to Identify Candidate Allosteric Sites on the Surface

3.1-a-ii Binding Leverage Calculations

3.1-a-iii Defining & Applying Thresholds to Select High-Confidence Surface-Critical Sites

3.1-a-iv Known Ligand-Binding Sites at the Surface

3.1-b Dynamical Network Analysis to Identify Interior-Critical Residues

3.1-b-i Network Formalism and Weighting Scheme

3.1-b-ii Decomposing Proteins into Modules Using Different Algorithms

3.1-c STRESS (STRucturally-identified ESSential residues)

3.2 High-Throughput Identification of Alternative Conformations

3.2-a Database-Wide Multiple Structure Alignments

3.2-b Identifying Distinct Conformations within a Multiple Structure Alignment

3.2-c Models of Conformational Change via Displacement Vectors from Alternative Conformations

3.2-c-i Inferring Protein Conformational Change Using Displacement Vectors from Alternative Conformations

3.2-c-ii Identifying Surface-Critical Residues Using Vectors from Alternative Conformations

3.2-c-iii Identifying Interior-Critical Residues Using Vectors from Alternative Conformations

3.2-c-iv Using Vectors from Alternative Conformations Recapitulates Results Using Normal Modes

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

3.3-a Conservation Across Species

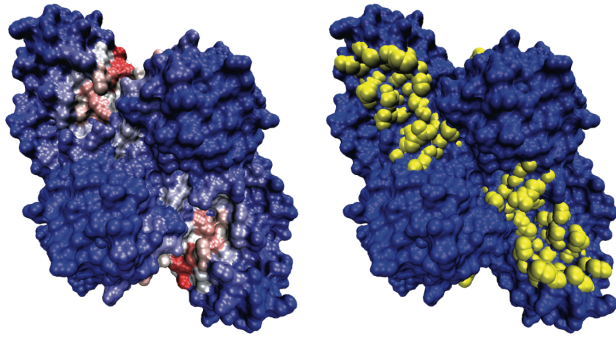
3.3-b Measures of Conservation Amongst Humans from Next-Generation Sequencing

4 – Supplemental References

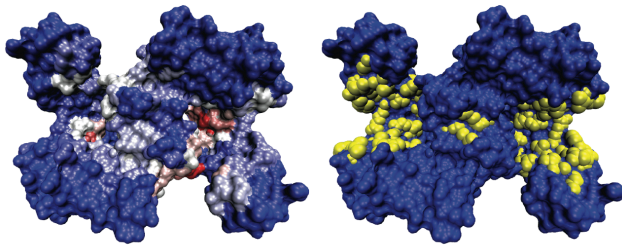
1 – Supplemental Figures

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

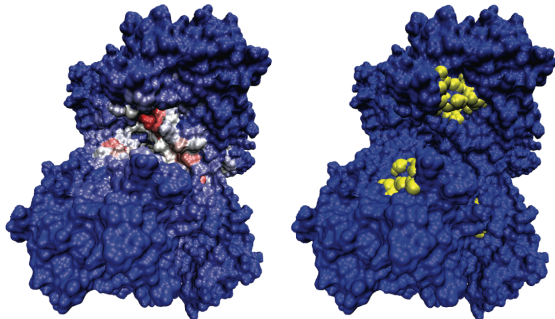
Below, each left image shows sites that are scored highly (i.e., surface-critical residues, in red), and each right image shows the residues (yellow) that actually come into contact of known ligands, based on the corresponding *holo* structure (see Table S1).



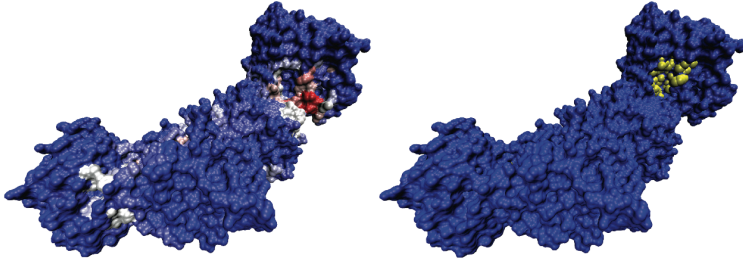
Phosphofructokinase (PDB ID 3pfk)



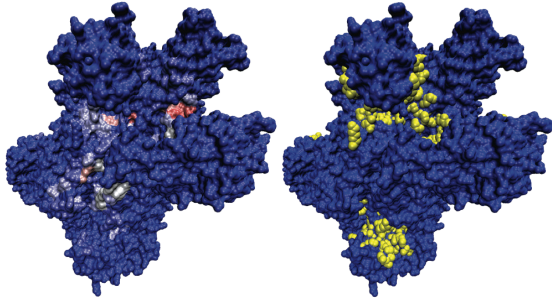
Adenylate Kinase (PDB ID 4ake)



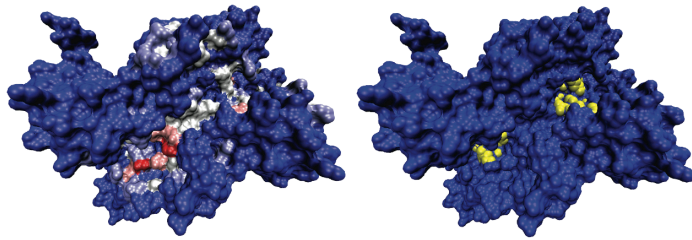
G6P-Deaminase (PDB ID 1cd5)



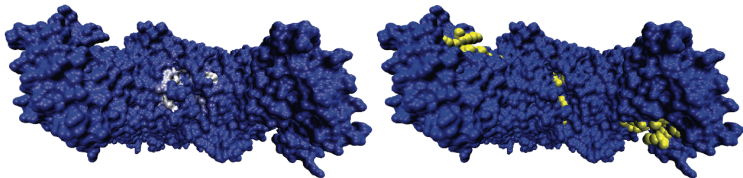
Trp Synthase (PDB ID 1bks)



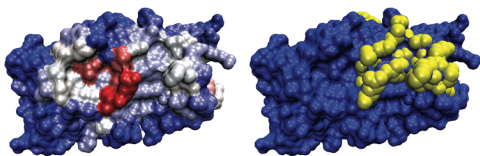
Glu Dehydrogenase (PDB ID 1nr7)



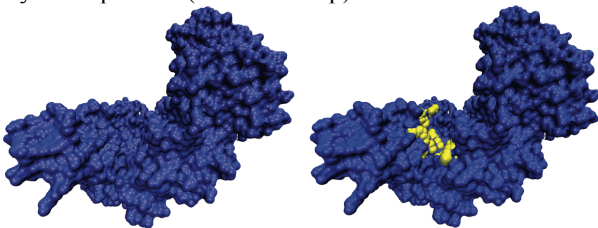
Thr Synthase (PDB ID 1e5x)



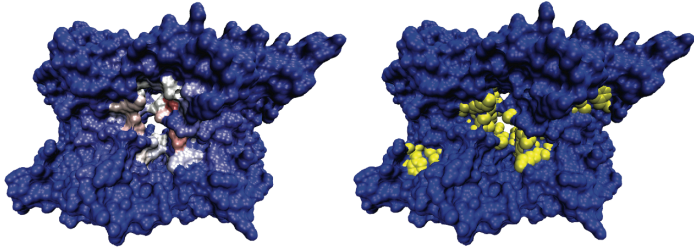
Malic Enzyme (PDB ID 1efk)



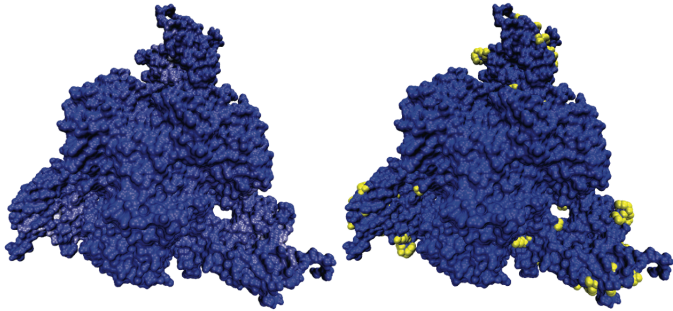
Tyr Phosphatase (PDB ID 2hnp)



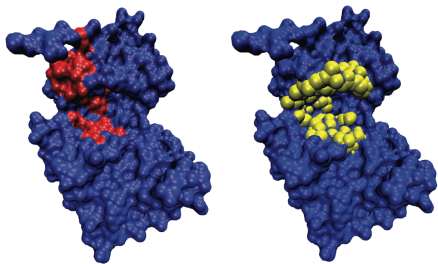
Arg Kinase (PDB ID 3ju5)



Phosphoribosyltransferase (PDB ID 1xtt)



Asp Transcarbamoylase (PDB ID 3d7s)



cAMP-dependent Kinase (PDB ID 1j3h)

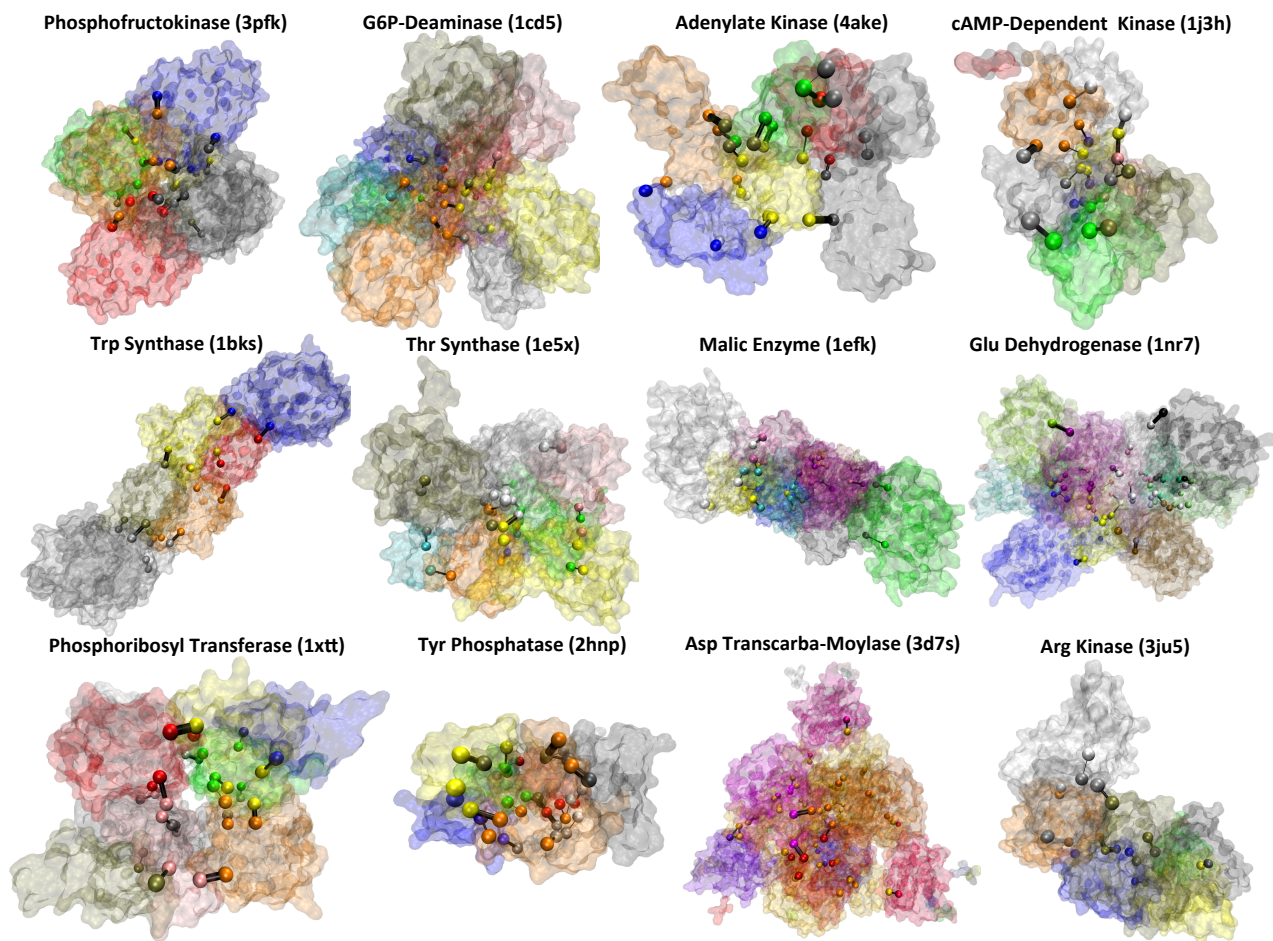
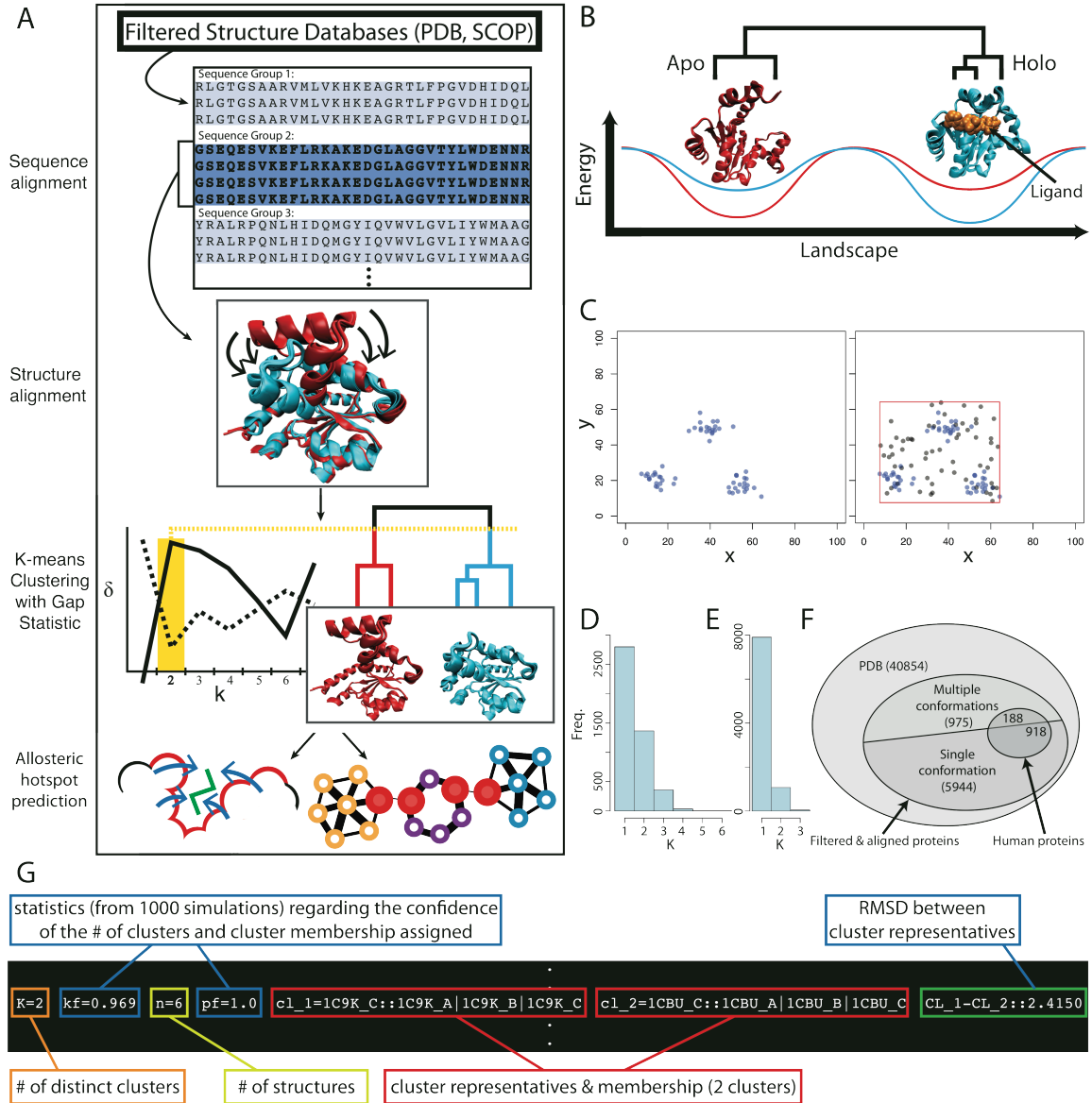


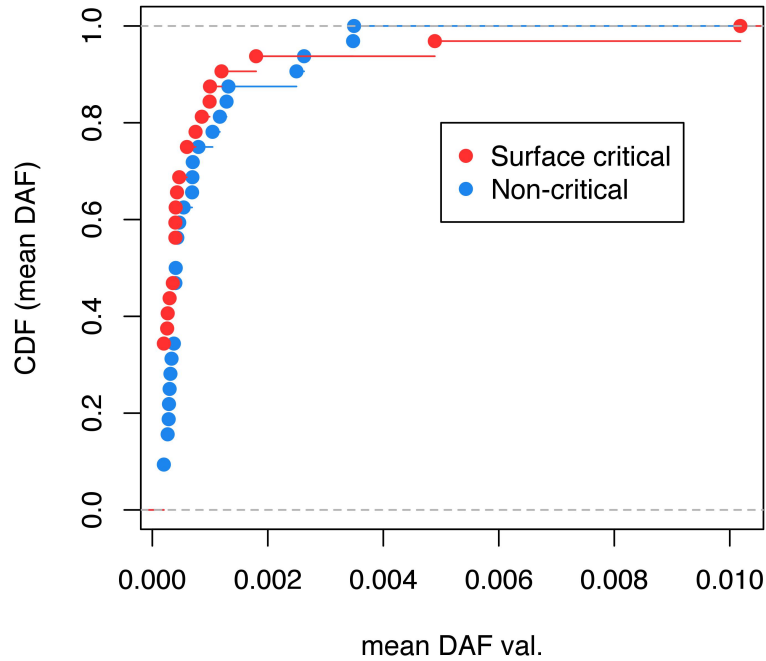
Figure S2: Community partitioning for canonical systems

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 in SI Methods section 3.1-b. 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.

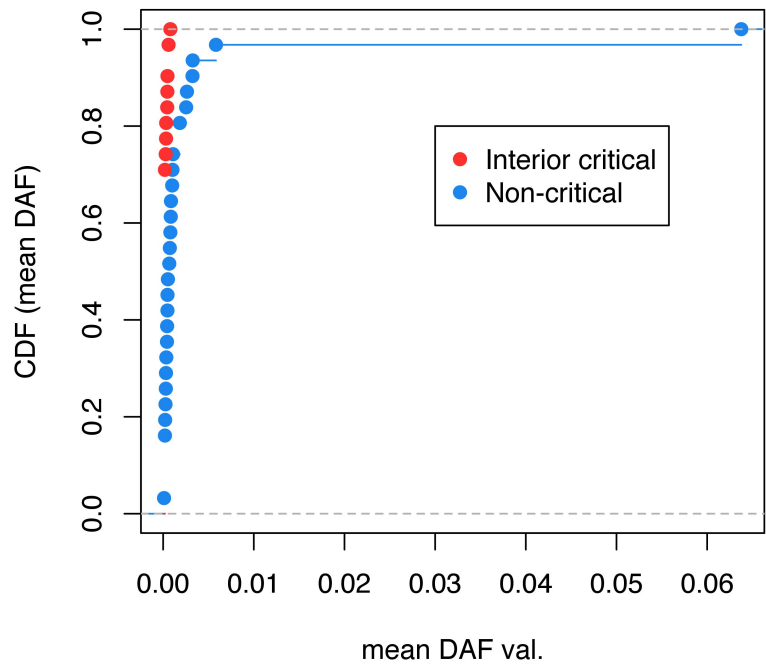
Figure S3: Pipeline for identifying alternative conformations throughout the PDB.

(A) 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 Figure 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 section 3.2-b 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. (B) 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: d3hpbq1, d3hpqa1, d2eckb1, d2ecka1, d1akeb1, and d1akea1. (C) 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. The distributions of the number conformations (i.e., “K”) for domains and chains are given in (D) and (E), respectively. Only proteins for which K exceeds 1 (for chains) are included in our dataset of multiple conformations. (F) Distinct proteins in our dataset within the context of high-quality X-ray structures in the PDB that we structurally aligned. A set of distinct proteins is such that no pair shares more than 90% sequence identity. (G) Figure 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 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 File S1 for further details.



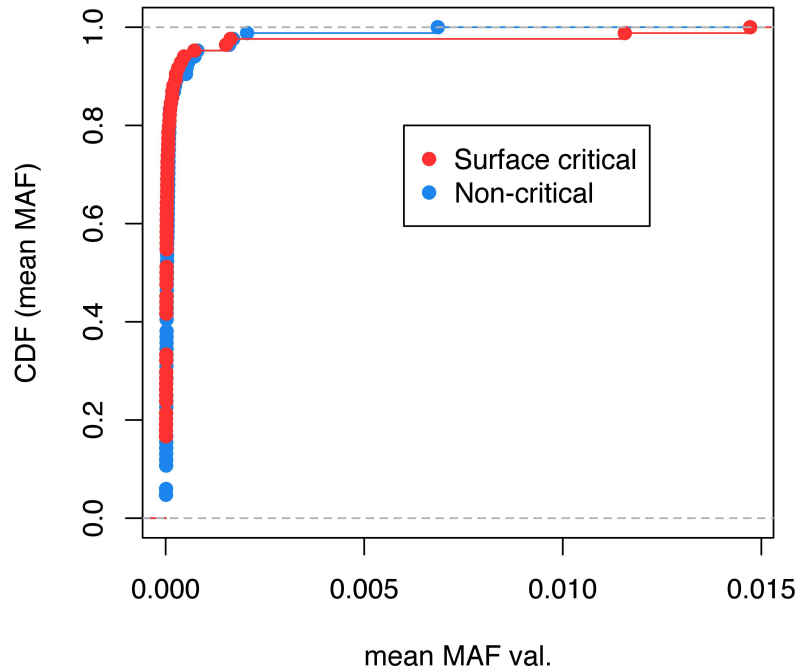


A) Cumulative distribution functions for mean DAF values of surface-critical and non-critical residues (p -val = 0.159, KS test)

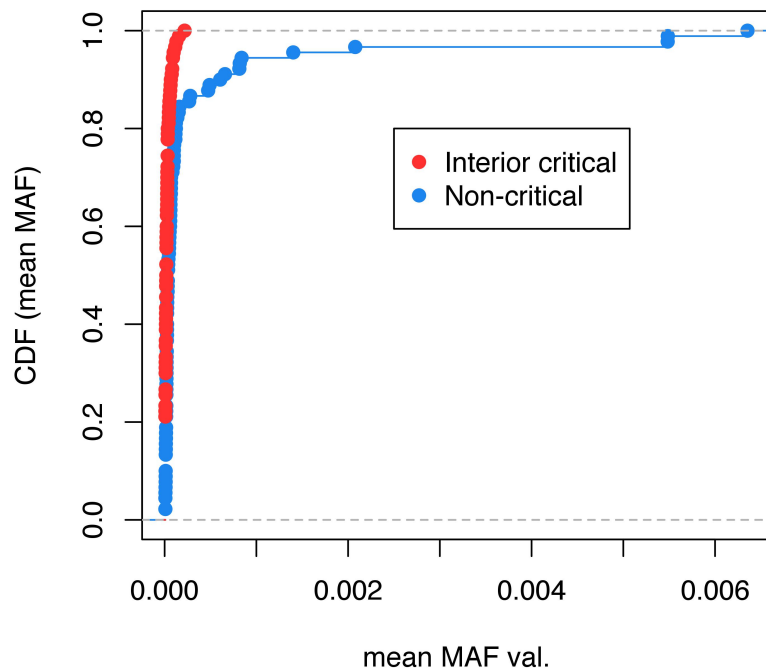


B) Cumulative distribution functions for mean DAF values of interior-critical and non-critical residues (p -val = 1.79×10^{-4} , KS test)

Figure S4: 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 ($p\text{-val} = 9.49e-2$, KS test)



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

Figure S5: Potential shifts in mean minor allele frequency distributions (in ExAC) using two-sample Kolmogorov-Smirnov tests

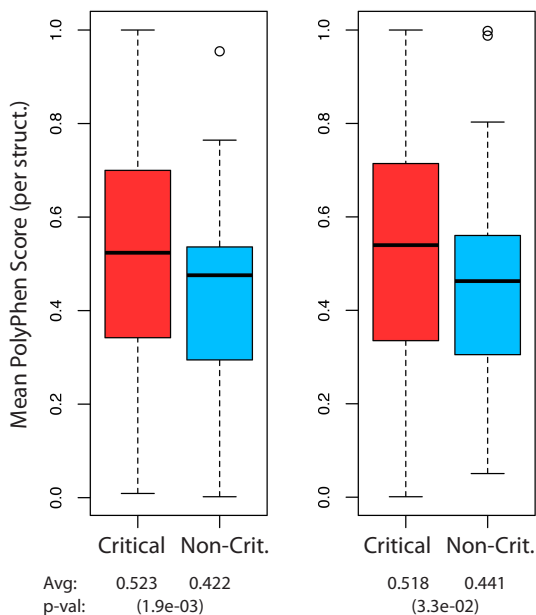


Figure S6: Mean PolyPhen scores for critical- and non-critical residues, as identified by ExAC
Left: Distributions (64 structures) of mean PolyPhen values on surface-critical residues (red) and non-critical residues (blue). *Right:* Distributions (70 structures) of mean PolyPhen values on interior-critical 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.

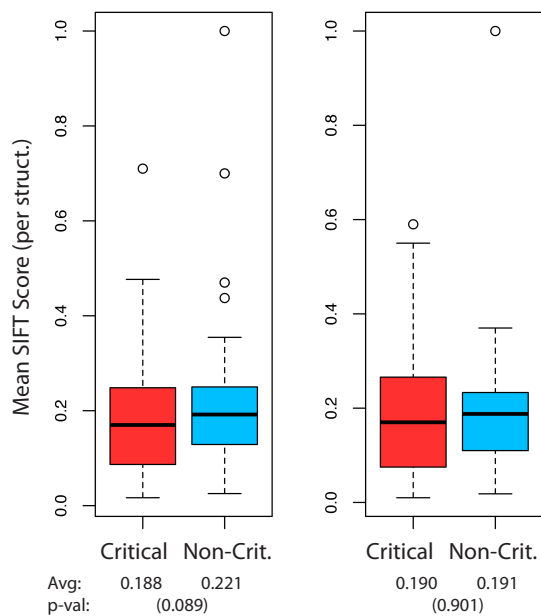


Figure S7: Mean SIFT scores for critical- and non-critical residues, as identified by ExAC
Left: Distributions (63 structures) of mean SIFT values on surface-critical residues (red) and non-critical residues (blue). *Right:* Distributions (65 structures) of mean SIFT values on interior-critical residues (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.

2 – Supplemental Tables

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, U5P)	1xtt (ACY, U5P)
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 (PO4)

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

These 12 proteins were chosen to constitute the canonical set for several reasons: the allosteric mechanisms of their natural ligands are well understood, and both the *holo* and *apo* states for each system are available and clearly distinguishable; in addition, these proteins have been extensively investigated in the contexts of both binding leverage and allostery in general. Ligands are given in parentheses (those in bold text designate the ligands used to define residues involved in ligand-binding interactions).

n	Mean fract. of ligand-binding sites captured
6	0.56
5	0.59
4	0.65
3	0.69
2	0.79
1	0.84

Table S2: 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 (n) is relaxed to lower values, 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 S3: 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.

3 - Supplemental Methods

3.1 Identifying Potential Allosteric Residues

Allosteric residues are predicted both on the surface and within the protein interior. In this study, these two sets of predicted allosteric residues are termed “surface-critical” and “interior-critical” residues, respectively. Notably, allosteric sites on the surface play mechanistic roles that are generally different from those within the interior: while surface sites often function as the source points or sinks of allosteric signals, the interior acts to transmit such information. Thus, different approaches are needed for selecting these two sets of residues. For both, biological assembly files from the PDB are used as the input to our analysis (Berman et al., 2000).

3.1-a Identifying Surface-Critical Residues

With the objective of identifying potential allosteric residues on the protein surface, we employ a modified version of the binding leverage method for identifying likely ligand binding sites (Figure 1A), as described previously (Mitternacht and Berezovsky, 2011). Allosteric signals may be transmitted over large distances by a mechanism in which the allosteric ligand has a global effect 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 (Fig. 1a). Such a modulation of the global motions may affect activity within sites that are distant from the allosteric ligand-binding site. We point the reader to work by Mitternacht and Berezovsky for a more detailed discussion regarding the binding leverage method (Mitternacht and Berezovsky, 2011), though a general overview of the approach, along with a detailed discussion of the changes we have implemented, are given below.

3.1-a-i Monte Carlo Simulations & Parameterization to Identify Candidate Allosteric Sites on the Surface

The surface of most proteins is a highly dense patchwork of pockets, ridges, protrusions, and clefts. Throughout this complex topology, there are many potential sites that may confer allosteric regulation upon binding by natural or artificial ligands. Thus, as a first step to identifying surface-critical sites, we aim to identify surface pockets that are capable of accommodating small ligands. These candidate allosteric sites are generated by Monte Carlo (MC) 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. The number of MC simulations is set to 10 times the number of residues in the protein structure, and the number of MC steps per simulation in our implementation is set to 10,000 times the size of the simulation box, 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.

Throughout the MC simulation, 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 atom’s distance to *alpha carbon atoms* in the protein – other heavy atoms or biophysical properties are not considered.

Our approach and set of applications differ from those previously developed in several key ways. When running MC simulations to probe the protein surface and generate candidate binding sites, we use 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.

The determination of how these parameters should be changed in an all-heavy atom model is fundamentally a problem of *optimization*. Thus, how are these parameters optimized in 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). **[Comment 2-2.2]** Specifically, the parameters to be optimized include (1) the ranges of favorable and unfavorable interactions (i.e., the *widths* of the potential function) and (2) the attractive and repulsive energies themselves (i.e., the *depths* and *heights* of the potential function).

For well *depths*, we tested models using several different attractive potentials, ranging from -0.05 to -0.75, including all intermediate factors of 0.05. For well *widths*, we first tried 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, in addition to sampling various well widths, we also performed the simulations using revised sets of distance cutoffs.

[Comment 2-2.1] Using this approach, the optimized set of parameters were as follows (here, $D_{lig-prot}$ designates the distance, in Angstroms, between a ligand atom and a protein atom):

<i>widths</i>	<i>depths & heights</i>
$\infty > D_{lig-prot} \geq 4.5:$	Energy = 0
$4.5 > D_{lig-prot} \geq 3.5:$	Energy = - 0.35 (attractive)
$3.5 > D_{lig-prot} \geq 3.0:$	Energy = +10 (repulsive)
$3.0 > D_{lig-prot} \geq 0.0:$	Energy = +10000 (strongly repulsive: effectively prohibited)

In addition to optimizing these parameters within the potential function, we also determined that setting the number of MC steps to 10,000 times the size of the simulation box (see above) provided the best convergence across multiple simulations on the same protein – that is, this number of steps better enabled us to recapture the same set of sites when running the simulations multiple times.

[Comment 2-3.1] After all candidate sites are identified by these MC simulations, pairs of sites with extremely high overlap are merged by combining any pair of sites that have a Jaccard similarity of at least 0.7, where the Jaccard similarity between sites i and j is $|i \cap j|/|i \cup j|$. After merging sites in this way, the residues of a given site are listed by their local closeness, and no more than 10 residues for a site are used. Local closeness (LC) is a geometric quantity that provides a measure of the degree of a residue in the residue-residue contact network; see (Mitternacht and Berezovsky, 2011b) for further discussion of LC. This entire process results in a list of sites on which binding leverage calculations can be performed.

3.1-a-ii Binding Leverage Calculations

Once candidate pocket sites are identified using the protocol outlined above, an obvious question is whether these sites can function allosterically by influencing global low-frequency motions of the protein. In order to rank the candidate sites by the degree to which they can impart such allosteric properties, the binding leverage associated with each candidate site is calculated.

First, normal modes analysis is applied to generate a model of the protein's low-frequency motions (alternatively, one may use direct displacement vectors between two structures; see SI Methods section 3.2-c). To generate these modes, we use the alpha carbon atoms in building the protein's elastic networks. Using default parameters, we take the top 10 (lowest-frequency) available non-trivial Fourier normal modes generated using the Molecular Modeling Toolkit (MMTK) (Hinsen, 2000). Specifically, these 10 low-frequency modes are produced using the "representative structures" within each cluster of a multiple structure alignment (for details on representative structures, see SI Methods section 3.2-b). Note that this exact same method for producing the modes was also used in the identification of interior-critical residues (see below).

Once the 10 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 (Figure 1A, top-right) receive a high score (termed the binding leverage for that site), whereas shallow sites with few interacting residues (Figure 1A, bottom-left) or sites that undergo minimal change over the course of a mode fluctuation (Figure 1A, bottom-right) receive low binding leverage scores. **[Comment 2-3.2]** Specifically, the binding leverage score for a given site is calculated as

$$\text{binding leverage} = \sum_{m=1}^{10} (\sum_i \sum_j \Delta d_{ij(m)}^2)$$

Here, the outer sum is taken over the 10 modes, and the pair of inner sums are taken over all pairs of residues (i,j) such that the line connecting the pair lies within 3.0 Angstroms of any atom within the simulated ligand. The value $\Delta d_{ij(m)}$ for each residue pair (i,j) represents the change in the distance between residues i and j when this distance is calculated using mode m . Thus, one may think of binding leverage as qualitatively predicting the extent to which a surface pocket is deformed when the protein undergoes conformational transitions.

3.1-a-iii Defining & Applying Thresholds to Select High-Confidence Surface-Critical Sites

As discussed in the main text, without applying thresholds 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 (Figure 2C). Thus, it is necessary to trim and process this list. To do so, we borrow from principles in energy gap theory (Bryngelson et al., 1995). Details regarding the calculations for establishing a threshold on the number of sites are given here.

For each of the N candidate binding sites in what we call “pre-processed ranked list of sites” (produced by the procedure outlined above), we calculate the value $\partial \text{BL} (j) / \Delta \text{BL}$. Here, j is the j^{th} site to appear in the pre-processed ranked list of sites, with this list of sites being ranked in descending order 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 defined as:

$$\Delta \text{BL} = \max_{\text{binding_leverage_score}} - \min_{\text{binding_leverage_score}}$$

ΔBL is thus the difference in the binding scores associated with the very top site and very bottom site in this ranked. 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 $\partial \text{BL} / \Delta \text{BL}$. 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 what we call 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

$$\text{redundancy_score}(i) = |\{R_{\text{site } i}\} \cap \{R_{\text{sites} > i}\}| / N_{\text{res } i}$$

Here, $\{R_{\text{site } i}\}$ is the set of residues in site i , $\{R_{\text{sites} > i}\}$ is the union of residues in all accepted sites above site i in the list of sites, $N_{\text{res } i}$ is the number of residues in site i , and the $|\dots|$ notation in the denominator of this ratio is meant to designate the number of residues in the indicated intersection. If this redundancy score is less than 0.4, then site i is considered to be truly distinct from all other sites, and it is included in the list of

distinct sites. If the redundancy score 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.

3.1-a-iv Known Ligand-Binding Sites at the Surface

Known ligand-binding residues of an *apo* structure are taken to be those within 4.5 Angstroms of the ligand in the corresponding *holo* structure (Table S1). Within the canonical set of proteins, surface-critical sites overlap with an average of 56% of the known-ligand binding sites (Table 1). It has previously been shown that the sites in aspartate transcarbamoylase (PDB ID 3D7S) are especially difficult to identify (Mitternacht and Berezovsky, 2011); excluding aspartate transcarbamoylase results in finding an average of 61% of known biological ligand binding sites. In addition, we emphasize that many of the “false positives” (sites predicted to be important allosterically, but do not correspond to known ligand binding sites) may nevertheless function as latent allosteric sites. Such sites potentially may impart allosteric properties through previously uncharacterized ligands or through artificial ligands (such as drugs targeted to specific proteins).

3.1-b Dynamical Network Analysis to Identify Interior-Critical Residues

As discussed, allosteric residues within the protein interior often act to transmit signals. The identification of such residues is accomplished by a network formalism (see Fig. 1B), wherein the objective is to identify network nodes (i.e., residues) that are essential for communication between communities (i.e., groups of highly inter-connected residues of the contact map). This first entails representing a protein structure as a network of interacting residues, and then weighing the connections (edges) between these residues using information about inferred protein motions. Once the edges are weighed, the network is decomposed into distinct modules, and the residues that are identified as being important for inter-module communication are identified as the interior-critical residues. The details of this formalism are provided here.

3.1-b-i Network Formalism and Weighting Scheme

The network representing interacting residues is first constructed. 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 within the network.

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 (as with the identification of surface-critical residues, it is also possible to model conformational changes by using information regarding pairs of distinct conformations; see the SI Methods section 3.2-c). Again, the 10 lowest-frequency non-trivial modes are produced using the “representative structures” (see discussion in SI Methods section 3.2-b) within each cluster of a multiple structure alignment for a given protein. We emphasize that, although ANMs are more coarse-grained than molecular dynamics, our use of ANMs is motivated by their much faster computational efficiency, which is a required feature for our database-scale analysis.

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

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

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 $\langle \mathbf{r}_i \bullet \mathbf{r}_j \rangle$ indicate that we are taking the mean value for the dot product $\mathbf{r}_i \bullet \mathbf{r}_j$ 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 Girvan-Newman (GN) algorithm (Girvan et al., 2002) 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, where a path length is the sum of its associated effective edge distances (see above). The shortest distance between each $\binom{N}{2}$ pair of nodes in the network of N residues is calculated with the Floyd-Warshall algorithm. See Figure S3 for examples of community partitions and associated interior-critical residues.

3.1-b-ii Decomposing Proteins into Modules Using Different Algorithms

We use the GN formalism to identify the community structure of networks as part of our framework to identify interior-critical residues. By identifying the “community structure”, 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. However, although we employ GN, many other algorithms have been devised to identify community structure.

In a comprehensive study comparing different algorithms (Lancichinetti and Fortunato, 2009), an information theory-based approach (Rosvall and Bergstrom, 2007) was shown to be one of the strongest methods. This approach (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 (Rosvall and Bergstrom, 2007).

Perhaps surprisingly, even though both GN and Infomap achieve similar network modularity (with GN being slightly better), 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 S3). 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.

3.1-c STRESS (STRucturally-identified ESSential residues)

We have developed an easy-to-use web tool in order to enable those in the structural biology community to identify surface- and interior-critical residues within their own proteins of interest. 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 (Figures 3 B,C). 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 (Figure 3D). 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).

3.2 High-Throughput Identification of Alternative Conformations

There are many proteins within the PDB for which multiple distinct conformations are available. In many cases, a large number of structures may represent a relatively small number of conformations. We have sought identify such alternative conformations using a structural clustering scheme as part of our framework for identifying critical residues. The purpose of developing this clustering scheme is three-fold:

- 1) We are interested in those structures that exhibit distinct conformations, as we are focusing on cases for which pronounced conformational change plays an essential role in allostery.
- 2) The clustering scheme ultimately enables us to perform an important control. Namely, it enables us to address the question: are the results robust to alternative methods of inferring information about conformational change? ANMs provide only one means of defining the vectors for predicted conformational change. However, another approach is to use the direct displacement vectors from the crystal structures of alternative conformations. This alternative constitutes a method that we term “absolute conformational change” (ACT) in the manuscript.
- 3) ANMs constitute the bulk of our analysis, so we must be confident that the structures analyzed are suitable: if a given protein is not believed to undergo significant conformational change, it may not be appropriate to apply ANMs, as the ANMs can incorrectly predict large-scale conformational change where no such change is believed to occur.

An overview of our pipeline is provided in Figure S9. Broadly, 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 models of protein conformational transitions to identify surface- and interior-critical residues.

3.2-a Database-Wide Multiple Structure Alignments

FASTA files of all SCOP domains were downloaded from the SCOP website (version 2.03) (Fox et al., 2014; Murzin et al., 1995). We first worked with domains to probe for intra-domain conformational changes, as better alignments are generally possible at the domain level. For all other analyses reported, all

results are based on groups of structures that are 100% sequence identical. We removed structures with resolution values poorer than 2.8, as well as any PDB files with R-Free values poorer than 0.28. STAMP (Russell and Barton, 1992) and MultiSeq (Roberts et al., 2006) were used to generate the multiple structure alignments (MSAs). For each MSA, the final output is a symmetric matrix representing all pairwise RMSD values, which are then used as input to the K-means module (below).

3.2-b Identifying Distinct Conformations within an MSA

For each MSA produced in the previous step, the corresponding matrix of pairwise RMSD values describes the degree and nature of structural heterogeneity among the crystal structures. The objective is to use this data in order to identify the biologically distinct conformations represented by an ensemble of structures. Our framework relies on a modified version of the K-means clustering algorithm, termed K-means clustering with the gap statistic (Tibshirani et al., 2001). *A priori*, performing K-means clustering assumes prior knowledge of the number of clusters (i.e., “K”) to describe a dataset, and the gap statistic enables one to identify the optimal number of clusters intrinsic to a complex or noisy dataset. Given multiple resolved crystal structures for a given domain, this method estimates the number of conformational states represented in the ensemble of structures.

As a first step toward clustering the structure ensemble of N structures, we use multidimensional scaling (MDS) to convert an N-by-N matrix of pairwise RMSD values into a set of N distinct points, with each point representing a structure in (N-1)-dimensional space. 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.

We point 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 (Tibshirani et al., 2001). However, an overview is provided here. Assume that the data takes the form of 60 data points, with each point represented in 2D space.

1) Start by assuming that the data can be represented with K clusters. Perform standard K-means clustering on the data 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 data, how well does this number of assigned clusters K actually represent the ‘true’ number of clusters, relative to a null model without any apparent clustering? To address this, produce a null distribution of 60 randomly (i.e., uniformly) distributed data points that lack any clear clustering such that the randomly placed points lie within the same bounding box of the observed data.

4) Repeat step (3) M times, and each time a random null distribution is produced, calculate $W_{null(K)}$ (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. The $\text{mean}_M\{\log(W_{null(K)})\}$ 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 (Tibshirani et al., 2001) for details:

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

5) Calculate the gap statistic $\delta(K)$, given K clusters. This measures 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. A high $\delta(K)$ signifies that our data is well-described using K clusters. Assuming K clusters, the gap statistic is given as:

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

6) Obtain successive values $\delta(K+1)$, $\delta(K+2)$, $\delta(K+3)$, etc. by incrementing the value for K and repeating the steps (1) - (5). The optimal K is the first (i.e., lowest) K such that $\delta(K) \geq \delta(K+1) - s_{k+1}$:

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

We confirmed that these K_{optimal} values accurately reflect the number of clusters by manually studying dozens of MSAs. We also examined several negative controls, such as CAP, an allosteric protein that does not undergo conformational change. We identified a vast array well-studied 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 (Figure S11). The gap statistic performed well in discriminating crystal structures that constitute such a diverse set.

Each structure is assigned to its respective cluster using the assigned optimal K -values as input to Lloyd's algorithm (i.e., standard K -means clustering). 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. This representative is the member with the lowest Euclidean distance to the cluster mean, using the coordinates obtained by MDS (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).

3.2-c Models of Conformational Change via Displacement Vectors from Alternative Conformations

[[Comment 2-6.2]]

Unless otherwise specified, we use normal modes analysis to model conformational change throughout this study. However, one potential concern with this approach is that normal modes may not faithfully represent plausible conformational changes. Thus, in order to determine whether or not the results are robust to different means of inferring motions (especially those results relevant to the conservation of critical residues), we also model conformational change using vectors connecting pairs of corresponding residues in crystal structures of alternative conformations. We term this approach "absolute conformational transitioning" (ACT). This more direct model of conformational change is especially straightforward to apply to single-chain proteins (applying ACT on a database scale to multi-chain complexes would introduce confounding factors related to chain-chain correspondence between such complexes when each complex has multiple copies of a given chain).

3.2-c-i Inferring Protein Conformational Change Using Displacement Vectors from Alternative Conformations

Given a particular protein, how are these ACT vectors defined in order to calculate critical residues? For demonstration, we discuss a hypothetical example consisting of a multiple structure alignment of 8 sequence-identical structures. Starting with the protein's multiple-structure alignment using all 8 structures, we determine the optimal number of clusters represented by the structure alignment using the K -means algorithm with the gap statistic (see the above SI Methods section 3.2-b). Suppose that these 8 structures may be grouped into 2 distinct clusters by our scheme (4 structures in *cluster A*, and 4 structures in *cluster B*, for instance). As discussed in SI Methods section 3.2-b, a representative structure is taken from each of these two clusters (*structure A* and *structure B*). These two representatives are taken to represent the alternative conformations for the protein. As an alternative to using ANMs, we may use *structure A* and *structure B* to try to infer information about the protein's global conformational shifts by assigning a displacement vector to each residue (for instance, residue Y140), where the displacement vector is simply defined by the two corresponding residues in the different structures within the structure alignment (i.e., Y140 within *structure A* of the structure alignment and Y140 within *structure B* of the structure alignment). Because the structure alignment was performed on sequence-identical structures, each residue in one of these two representative structures matches a corresponding residue on the other representative structure. If each of the two structures represents a sequence-identical protein consisting of

200 residues, then 200 ACT vectors are drawn in order to represent the conformational change in transitioning from one conformation to the other. These 200 ACT vectors for the protein may then be used to identify surface- and interior-critical residues (see below), and downstream analysis on these residues is then performed.

3.2-c-ii Identifying Surface-Critical Residues Using Vectors from Alternative Conformations

All preliminary steps performed when identifying surface-critical residues using normal modes (such as the MC search) are the same as those when using ACT vectors, with the important difference, of course, being the use of these ACT vectors as oppose to using eigenvectors when inferring motion.

[Comment 2-3.2] Thus, when using ACT vectors, the binding leverage score for a given site is simply calculated as:

$$\text{binding leverage} = \sum_i \sum_j \Delta d_{ij}^2$$

where the sum is taken over all pairs of residues (i,j) such that the line connecting the pair lies within 3.0 Angstroms of any atom within the simulated ligand, and the value Δd_{ij} for each residue pair (i,j) represents the change in the distance between residues i and j when this distance is calculated in alternative crystal structure. Thus, for each residue, the 10 vectors provided by the normal modes are simply replaced by the single ACT vector that defines the change in position of that residue when going from the protein conformation given by one representative structure to the conformation given by the other representative.

3.2-c-iii Identifying Interior-Critical Residues Using Vectors from Alternative Conformations

When identifying interior-critical residues, ACT vectors may be produced in the exact same way that they are produced when identifying surface-critical residues. When identifying interior-critical residues, the inferred conformational changes are used in order to assign weights within the residue contact maps. In the scheme in which normal modes are used, these weights are assigned by averaging over to 10 sets of vectors given by the 10 modes. However, when using ACT vectors, there is only one vector for each residue (i.e., the vector defining the “displacement” defined by two structures). Thus, when using ACT vectors, the weight parameters are calculated as

$$C_{ij} = \text{Cov}_{ij} / \sqrt{(|\mathbf{r}_i|^2 * |\mathbf{r}_j|^2)}$$

where

$$\text{Cov}_{ij} = \mathbf{r}_i \bullet \mathbf{r}_j$$

Here, \mathbf{r}_i denotes the vector that defines the change in position for residue i when going from one representative conformation to the other.

3.2-c-iv Using Vectors from Alternative Conformations Recapitulates Results Using Normal Modes

When we use ACT vectors to apply the modified binding leverage framework for these proteins, we again observe that our surface-critical residues are significantly more conserved than are non-critical residues (Figure 6A), and the same trend is also observed when ACT vectors are applied in our dynamical network analysis for identifying interior-critical residues (Figure 6B). The fact that ACT vectors produce a similar set of results to those obtained using normal modes analysis suggests that our approach is robust to different methods for inferring protein conformational change. We note that 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.

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

How conserved are the surface- and interior-critical residues identified, relative to other residues in the protein? Certainly, allosteric residues are known to exhibit conservation, and we should expect that the critical residues identified exhibit strong conservation. Conservation may be measured across diverse evolutionary time scales. Metrics for selective constraint that correspond to long evolutionary time scales entail sequence comparisons across diverse species. At the other extreme, metrics for short-term evolutionary conservation entail analyzing multiple genomes from within the same species (e.g., multiple human genomes). In order to evaluate the relative conservation of the critical residues identified in this study, we measure conservation using both types of measures, and demonstrate that, as expected, critical residues are under stronger evolutionary constraint relative to other regions of the protein.

3.3-a Conservation Across Species

All cross-species conservation scores represent the ConSurf scores, as downloaded from the ConSurf Server (Ashkenazy et al., 2010; Celniker et al., 2013; Glaser et al., 2003; Landau et al., 2005), in which ConSurf scores for each protein chain are normalized to have a mean ConSurf score of 0 [Comment 2-7] (the ConSurf score variance is 1 for each chain). 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., Figures 3B and 3F, and Figure 6) 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.

3.3-b 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 (McVean et al., 2012). VCF

files containing the annotated variants were generated using VAT (Habegger et al., 2012). 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 (Smedley et al., 2015). 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. 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., Figures 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 Figures 5A and 5C. 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 Figures 5B and 5D.

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 (Khurana et al., 2013; McVean et al., 2012; Tennessen et al., 2012).

4 - Supplemental References

- Altschul, S.F., Madden, T.L., Schäffer, A.A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D.J. (1997). Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* *25*, 3389–3402.
- Ashkenazy, H., Erez, E., Martz, E., Pupko, T., and Ben-Tal, N. (2010). ConSurf 2010: calculating evolutionary conservation in sequence and structure of proteins and nucleic acids. *Nucleic Acids Res.* *38*, W529–W533.
- Bryngelson, J.D., Onuchic, J.N., Succi, N.D., and Wolynes, P.G. (1995). Funnels, pathways, and the energy landscape of protein folding: A synthesis. *Proteins Struct. Funct. Bioinforma.* *21*, 167–195.
- Celniker, G., Nimrod, G., Ashkenazy, H., Glaser, F., Martz, E., Mayrose, I., Pupko, T., and Ben-Tal, N. (2013). ConSurf: using evolutionary data to raise testable hypotheses about protein function. *Isr. Journal Chem.* *13*, 199–206.
- Fox, N.K., Brenner, S.E., and Chandonia, J.-M. (2014). SCOPe: Structural Classification of Proteins--extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic Acids Res.* *42*, D304–D309.
- Glaser, F., Pupko, T., Paz, I., Bell, R.E., Bechor-Shental, D., Martz, E., and Ben-Tal, N. (2003). ConSurf: identification of functional regions in proteins by surface-mapping of phylogenetic information. *Bioinformatics* *19*, 163–4.
- Habegger, L., Balasubramanian, S., Chen, D.Z., Khurana, E., Sboner, A., Harmanci, A., Rozowsky, J., Clarke, D., Snyder, M., and Gerstein, M. (2012). VAT: a computational framework to functionally annotate variants in personal genomes within a cloud-computing environment. *Bioinformatics* *28*, 2267–2269.
- Hinsen, K. (2000). The Molecular Modeling Toolkit: A New Approach to Molecular Simulations. *J. Comput. Chem.* *21*, 79–85.
- Hubbard, S. J., and Thornton, J. M. (1993). Naccess. Computer Program, Department of Biochemistry and Molecular Biology, University College London, 2(1).
- Lancichinetti, A. and Fortunato, S. (2009). Community detection algorithms: a comparative analysis. *Phys Rev E Stat Nonlin Soft Matter Phys.* *80*, 56117.
- Landau, M., Mayrose, I., Rosenberg, Y., Glaser, F., Martz, E., Pupko, T., and Ben-Tal, N. (2005). ConSurf 2005: the projection of evolutionary conservation scores of residues on protein structures. *Nucleic Acids Res.* *33*, W299–W302.
- Mitternacht, S. and Berezovsky, I.N. (2011b). A geometry-based generic predictor for catalytic and allosteric sites. *Protein Eng Des Sel* *24*: 405–409.
- Murzin, A.G., Brenner, S.E., Hubbard, T., and Chothia, C. (1995). SCOP: A structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.* *247*, 536–540.
- Sokal, R.R. (1958). A statistical method for evaluating systematic relationships. *Univ. Kans. Sci. Bull.* *38*, 1409–1438.
- Roberts, E., Eargle, J., Wright, D. and Luthey-Schulten, Z. (2006). MultiSeq: unifying sequence and structure data for evolutionary analysis. *BMC Bioinformatics* *7*, 382.
- Rosvall, M. and Bergstrom, C.T. (2007). An information-theoretic framework for resolving community structure in complex networks. *Proc. Natl. Acad. Sci. U. S. A.* *104*, 7327–7331.
- Russell, R.B. and Barton, G.J. (1992). Multiple protein sequence alignment from tertiary structure comparison: assignment of global and residue confidence levels. *Proteins* *14*, 309–323.
- Smedley, D., Haider, S., Durinck, S., Pandini, L., Provero, P., Allen, J., Arnaiz, O., Awedh, M.H., Baldock, R., Barbiera, G., et al. (2015). The BioMart community portal: an innovative alternative to large, centralized data repositories. *Nucleic Acids Res.* *43*, W589–W598.
- Tennessen, J.A., Bigham, A.W., O'Connor, T.D., Fu, W., Kenny, E.E., Gravel, S., McGee, S., Do, R., Liu, X., Jun, G., et al. (2012). Evolution and functional impact of rare coding variation from deep sequencing of human exomes. *Science* *337*, 64–9.
- Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *J. R. Stat. Soc.* *63*, 411–423.