Chapter 5

Computationally Assisted Engineering of Protein Cages

Maziar S. Ardejani and Brendan P. Orner

Abstract

A hybrid computational method incorporating topographic analysis of protein surfaces and free-energy calculations of protein-protein interactions in protein nanocages is described. This design strategy can be used to engineer protein cages for enhanced structural stability and assembly.

Key words Protein engineering, Computational design, Protein-protein interactions, Ferritins

1 Introduction

Many self-assembling protein cages have evolved to selectively isolate biological material, such as mineralized iron in the case of the ferritins or genetic material within virus capsids, and release it on demand. Due to these encapsulation properties in addition to their nanoscale, protein cages are being harnessed increasingly for a wide range of nanobiotechnological applications from drug delivery to nanomaterials and catalysis [1]. However these applications are often limited by the non-ideal physical properties of these protein complexes. Therefore, it may be desirable to reengineer protein cages for enhanced utility. However, complete de novo rational design of complex folded proteins is extremely difficult [2], and this challenge is compounded for self-assembling nanoscaled protein quaternary structure [3]. Thus, the development and implementation of tools for the computationally assisted manipulation of protein-protein interactions involved in self-assembly, especially those generating nanostructured protein complexes, are of great recent interest [4]. Here, we describe a hybrid computational method incorporating topographic analysis of protein surfaces and free-energy calculations of protein-protein interactions to enhance the structural energetics of a protein nanocage.

Protein nanostructure is often assembled from multiple protein chains and is therefore controlled by different protein-protein interfaces, the number and nature of which depend on

Brendan P. Orner (ed.), *Protein Cages: Methods and Protocols*, Methods in Molecular Biology, vol. 1252,
DOI 10.1007/978-1-4939-2131-7_5,

stoichiometry, symmetry, and similarities between the monomers. Thus, to stabilize oligomeric protein complexes, an obvious strategy would be to manipulate interactions at the protein-protein interfaces directly [5]. However, the direct strategy ignores the fact that the hierarchical interdependence of the interactions made at these interfaces often complicates the redesign process through positive and negative cooperativity [4]. In other words, unfavorable structural deviations caused by small atomic clashes at lower-stoichiometry interfaces can be amplified through the structural hierarchy, resulting in complete disruption of higher-order assembly even if the specific redesigned interface is enhanced [6]. Thus, current protein design methodology is limited when applied to engineering nanostructures, and it rarely can be applied to multi-subunit, self-assembling systems [5]. Hence, improving the accuracy and predictive power of the currently available computational methodology is desirable. One strategy to do this could be the development of new hybrid computational approaches.

Proteins typically fold into well-packed three-dimensional structures. However, some oligomeric proteins, for various functional and non-functional reasons, possess protein-protein interfaces interrupted by solvent-accessible "concavities" which are further characterized by the number of openings they provide to bulk solvent: pockets (one opening) and pores (two openings). For example, some ferritins have concavities near their twofold, threefold, fourfold, and "B-site" interfaces (*see* Fig. 2 for a graphical definition of these interfaces). Some of these concavities are thought to be involved in the traffic of chemical species required for iron biomineralization inside the cage [7, 8].

In most protein design strategies, solvent-accessible concavities are rarely distinguished from other "packing imperfections" although in a few examples, with monomeric proteins, cavity-filling or cavity-making mutations have been used to probe folding [9–17]. The structural analysis program CASTp differentiates packing imperfections at protein-protein interactions based on their solvent accessibility [18]. The structural distinctness of these imperfections, which can be analytically detected by CASTp, has been exploited to design new molecular interactions through the optimization of side-chain packing in protein-protein interfacial concavities [6, 19].

In this chapter, we describe a method that we have successfully employed to stabilize [6] and enhance the self-assembling properties [19] of a protein nanocage, *E. coli* bacterioferritin. Our previous analyses have shown that most stabilizing mutations were associated with interfacial concavities; thus we reasoned that an efficient computational search could be achieved by limiting our focus to optimizing pocket-associated residues rather than the entire interface. In addition, we thought that the assembly state would be more tolerant to mutations within concavities because

there is more volume within which to generate new interactions [6]. Moreover, this method finds mutations that either block the interfacial pores or reduce the volume of the pockets, each of which effects is respectively suitable for making the cage impermeable and/or enhancing its stability.

The procedure described here is general and should be easily ported to other ferritins and protein cages. Our approach has been applied to single-point mutations but could be adapted to larger, more extensive studies. Furthermore, our suggestions regarding filtering out potential problematic residues and the number of structures to carry forward are based on developing a manageable project for a single PhD student in a modestly funded lab.

2 Materials

PDB file for a high-resolution structure of the protein cage [20] (downloadable at: http://www.rcsb.org/pdb/).

Structure visualization program (e.g., UCSF Chimera [21], downloadable at: http://www.cgl.ucsf.edu/chimera/).

FoldX program [22] (downloadable at: http://foldx.crg.es/).

CASTp [23] (web server accessible at http://cast.engr.uic.edu/).

BLAST [24] (user interface: http://blast.ncbi.nlm.nih.gov/).

Sequence alignment program (for example, ClustalX [25] downloadable at: http://www.clustal.org/).

3 Methods

This approach to engineer enhanced protein cage stability and self-assembly is based on a hybrid computational method incorporating topographic analysis of protein surfaces and free-energy calculations of protein-protein interactions. A summary of this approach is shown in the workflow diagram (*see* Fig. 1). This approach requires a protein data bank file (.pdb) containing the high-resolution structure of the protein cage of interest. These can be obtained from the RCSB Protein Data Bank. To prepare the structure for the calculations, it first must be simplified through the generation of an oligomeric subcomplex focusing on the protein-protein interaction of interest. The energy of this subcomplex is then computationally minimized to generate a structure with slightly altered atomic coordinates of the side chains. The virtual screening for stabilizing mutations is initiated by the identification of pockets at the oligomeric interfaces through surface topographic analysis of the minimized subcomplexes. Saturation mutagenesis is next performed through a series of free-energy calculations. Because these are low-level calculations, it is essential to

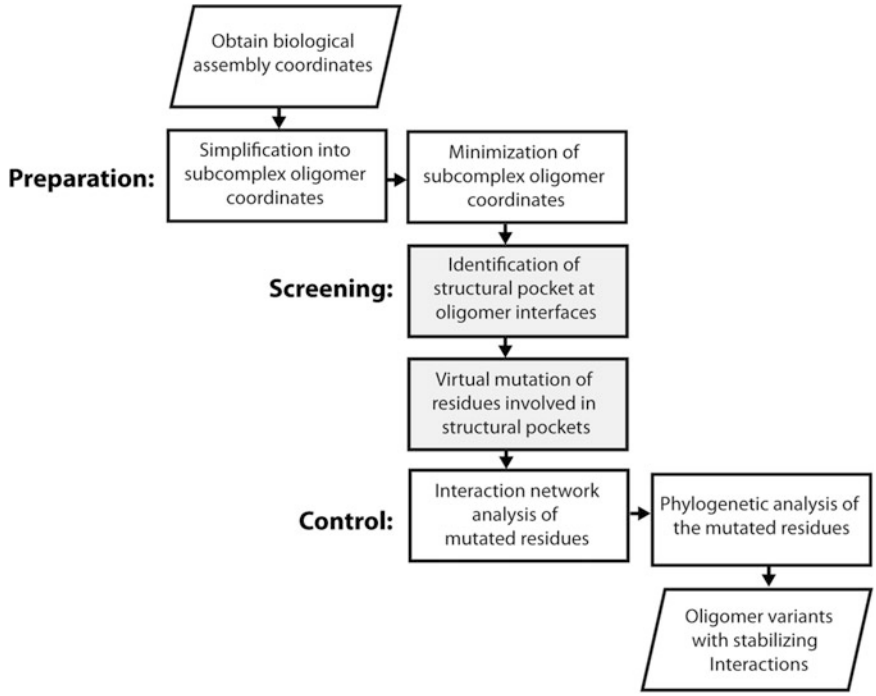

Fig. 1 A minimal computational workflow to design stabilized protein cages. Three stages of our strategy consist of: (1) preparation of the initial structure file into a minimized subcomplex, (2) virtual screening for stabilizing mutations using a minimal computation approach employing a hybrid of surface topographic analysis of protein-protein interfaces and free-energy simulation of their interactions, and (3) virtual control experiments to filter out potentially unreliable hits

perform replicates and virtual control experiments to improve the success rate. Therefore, interaction network and phylogenetic analyses are then performed on the predicted stabilizing mutations to ensure they do not impose any potentially destabilizing effects. Only variants that pass these controls are taken on for cloning, expression, and in vitro analysis.

3.1 Preparation of the Cage Subcomplex Structures

1. Retrieve the atomic coordinates of the protein cage either generated de novo or downloaded from the RCSB Protein Data Bank (*see* **Notes 1** and **2**).
2. Use the visualization program to extract the coordinates of an oligomeric subcomplex of the cage from the PDB file. A subcomplex oligomer (e.g., dimer, trimer, etc.) is created through deletion of all except the protein chains that comprise the protein-protein interface of interest. As most protein cages are symmetric, this simplification will reduce the computation time while causing little loss in accuracy (*see* **Note 3**).
3. Use the ⟨RepairPDB⟩ command in the FoldX software to perform an initial minimization of the subcomplex (at 298 K, pH 7, 0.05 M ionic strength; *see* **Note 4**).

3.2 Screening for Stabilizing Mutation by Virtual Saturation Mutagenesis of Pocket-Forming Residues

1. Submit the repaired subcomplex to CASTp [23] for detection and analysis of the pockets. Visualize the CASTp results using the visualization program. The CASTp calculations will detect many pockets and pores in the structure of each subcomplex. Rank them based on the molecular volume and visually confirm that these concavities are in the protein-protein interface of interest and not involved with protein monomers located outside the subcomplex (*see* **Note 5**).
2. Using the visualization program, export the list of residues involved in the selected pocket (e.g., use the ⟨write list⟩ command in Chimera).
3. Virtually mutate all of the residues involved in the interfacial concavity to all other amino acids using the ⟨BuildModel⟩ command of FoldX. This will generate an output file whose name begins with "Average_" Save this file for further analysis (*see* **Note 6**).
4. Open the resulting "Average_" output file using a spreadsheet editor and sort the data in ascending order by "total energy" to rank the mutations based on their degree of stabilization. Select the five most stabilizing mutations for further analysis (*see* **Note 7**).

3.3 Structural Analysis as Virtual Control Experiments to Filter Out Potential False Positives

1. Generate the minimized structure of the subcomplex for each of the selected mutants using the ⟨BuildModel⟩ command in FoldX and setting the "OutPDB" value to "true"
2. Submit each of these minimized structures to CASTp to analyze the structural effect of the mutation on the targeted concavity. This analysis will determine if the mutation indeed reduces the volume of the concavity as expected (*see* **Note 8**).
3. Use the ⟨PrintNetworks⟩ command in FoldX to extract the interaction network information from the minimized subcomplex structures of WT and the mutants. Identify mutated residues and compare the WT with the mutants to determine if new interactions across the interface were generated (*see* **Note 9**).
4. Obtain the amino acid sequence of homologous proteins from BLAST and perform an alignment to make sure that the mutated residues are not highly conserved (*see* **Note 10**).
5. Select three mutants that pass the control filters in the last three steps and assess them in vitro for stability.

4 Notes

1. If downloading the structure from the PDB, be sure to select the "biological assembly" as this will usually be the entire oligomeric cage as opposed to a single monomer.

2. Having a high-quality structure of the cage (resolution below 2.2 Å) is essential for successful analysis. Structures with a resolution below 2.8 Å can also be considered, but the computational results will be less reliable, and the software may have problems with some high-energy side-chain conformations.
3. A typical protein cage is composed of tens of thousands of atoms. Performing molecular-modeling calculations for such a large system requires tremendous amounts of computational power and time, making the task unfeasible. Our approach to break down large cage complexes into smaller oligomeric subcomplexes reduces the computational demand. For example, in the octahedral, homo-24-meric cage of a maxiferritin (*see* Fig. 2), each monomer interacts with six surrounding monomers through different types of interfaces, some of which are symmetry related. That is, in each ferritin cage, there are 6 tetramerization sites surrounding the fourfold axes, 8 trimerization sites on the threefold axes, 12 dimerizing interactions at the twofold axes of symmetry, and 24 "B-site interfaces". A combination of these interfaces constitutes the complete set of protein-protein contacts made by a monomer in the cage oligomerization state (e.g., the black monomer in Fig. 2).

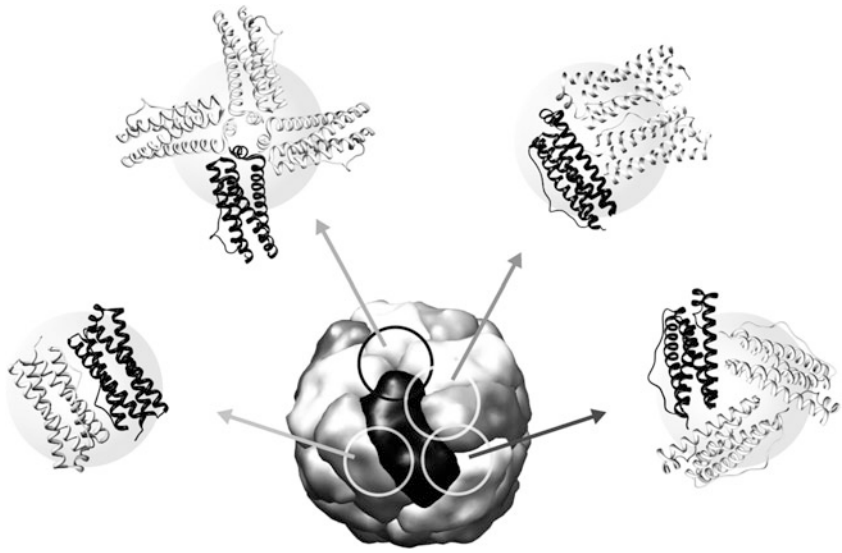

Fig. 2 Four different types of representative subcomplex oligomers into which a typical maxiferritin nanocage with octahedral, 432 symmetry can be broken down to reduce computational demand when analyzing their protein-protein interfaces. Each monomer (in *black*) interacts with six surrounding monomers through different types of interfaces, some of which are symmetry related. Clockwise, starting on the *left*: in each ferritin cage, there are 12 dimerizing interactions at the twofold axes of symmetry, 6 tetramerization sites surrounding the fourfold axes, 24 "B-site" interfaces, and 8 trimerization sites on the threefold axes. In order to ensure that interfacial concavities are analyzed accurately, some of these subcomplexes may be symmetry redundant. For example, in the maxiferritin, we define the "B-site interface" for the calculations although formally it is defined by combinations of the other interfaces (*see* **Note 3**)

4. A prerequisite for the computational analysis is to have a good structure, that is, a protein with no van der Waals clashes or no residues with high-energy rotamers. Also it is necessary to make sure that there are no missing atoms or any gaps in the backbone. To fulfill these requirements the ⟨RepairPDB⟩ command in FoldX is used to prepare PDB files for subsequent calculations. During this process, residues that have unfavorable torsion angles, van der Waals clashes, etc. are identified and optimized to find new energy minima. In the FoldX protocol, the side chains are rotated, while the backbones are fixed, and buffer salts and unwanted ligands are removed. These "repaired" PDB files are then used to perform surface topography analysis using CASTp in the next step.
5. Concavities detected by CASTp on the edges of a subcomplex are less useful because the effect of the missing monomers is neglected during their detection. With that said, this emphasizes the importance of analyzing a number different protein-protein interactions through the selection of different subcomplexes. Thus, some of these subcomplexes may be symmetry redundant. For example, in the maxiferritin, we define a "B-site interface" for the calculations although formally combinations of the other interfaces define it.
6. To save hard drive space, set the "OutPDB" value to "false" in the virtual screening step. Later in the control step, we will set this value to "true" in order to generate PDB files for the minimized structure of only the selected mutants.
7. It has been previously shown that methionine side chains in a protein can accommodate many different geometries and thus provide a considerable degree of conformational flexibility resulting in the lack of specificity for methionine-rich protein surfaces [26, 27]. Moreover, the oxidation of methionine, which occurs in a wide variety of conditions, has been shown to cause structural instability and changes in the aggregation states of proteins [28]. Thus, we generally remove Met from the list of viable stabilizing designs. In addition, mutations to proline can also be problematic due to loss of a backbone hydrogen bond and backbone flexibility effects. Because FoldX assumes a rigid backbone, we typically remove proline mutations from candidate designs.
8. In the *E. coli* bacterioferritin cage for instance, there is a large concavity (*see* Fig. 3) at the twofold dimeric interface which contracts upon computationally predicted replacement of Asn^{23} with hydrophobic amino acids. We have shown in vitro that these mutations enhance the thermal stability of the ferritin [6].
9. If the WT residue that was mutated has a large number of contacting atoms (more than 20), it is considered "highly buried."

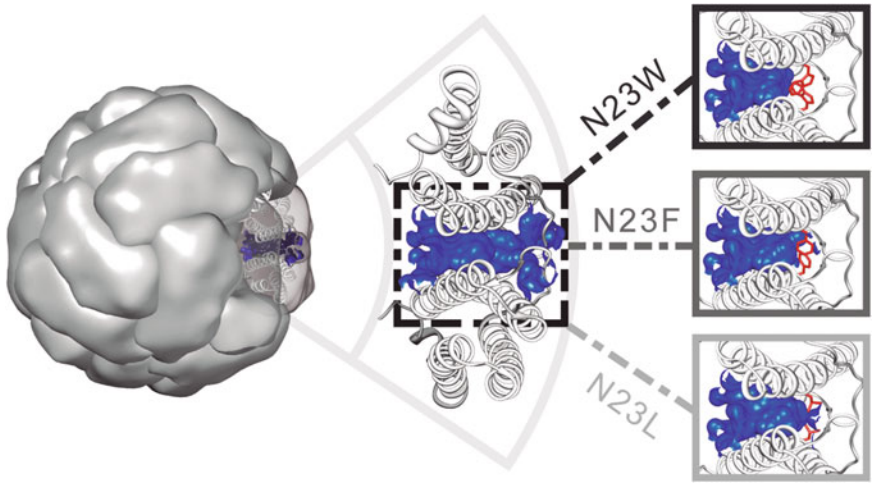

Fig. 3 Computationally predicted mutation of Asn^{23} to hydrophobic residues causes the large structural concavity (depicted as a *blue surface*) at the twofold dimeric interface of *E. coli* bacterioferritin to contract. The figure is adopted from Ardejani et al. [6] with permission

This may indicate that it is making structurally important interactions (as opposed to interactions with bound solvent within the concavity). In these cases, we often choose to avoid mutating these residues. This consideration is especially critical if the residue is involved in hydrogen bonding. We have noticed that the energetic penalty implemented in the FoldX energy function for removing a hydrogen bond can be understated compared to the other energy terms. Thus, the software may often falsely predict that a mutation, which removes a WT hydrogen bond, to have a stabilizing effect on the complex. Therefore, special care should be taken when residues involved in hydrogen bonds are mutated especially if the hydrogen bonds are part of a large interaction network.

10. A high level of conservation can give an indication that the residue is structurally or functionally important. Thus, avoiding mutation of these might be recommended, depending on the goals of the engineering exercise.

References

1. Uchida M, Klem MT, Allen M, Suci P, Flenniken M, Gillitzer E, Varpness Z, Liepold LO, Young M, Douglas T (2007) Biological containers: protein cages as multifunctional nanoplatforms. Adv Mater 19:1025–1042
2. Dahiyat BI, Mayo SL (1997) De novo protein design: fully automated sequence selection. Science 278:82–87
3. Ardejani MS, Orner BP (2013) Obey the peptide assembly rules. Science 340:561–562
4. Grigoryan G, Kim YH, Acharya R, Axelrod K, Jain RM, Willis L, Drndic M, Kikkawa JM, DeGrado WF (2011) Computational design of virus-like protein assemblies on carbon nanotube surfaces. Science 332:1071–1076
5. King NP, Sheffler W, Sawaya MR, Vollmar BS, Sumida JP, André I, Gonen T, Yeates TO, Baker D (2012) Computational design of self-assembling protein nanomaterials with atomic level accuracy. Science 336:1171–1174

6. Ardejani MS, Li NX, Orner BP (2011) Stabilization of a protein nanocage through the plugging of a protein–protein interfacial water pocket. Biochemistry 50:4029–4037
7. Yao H, Wang Y, Lovell S, Kumar R, Ruvinsky AM, Battaile KP, Vakser IA, Rivera M (2012) The structure of the BfrB–Bfd complex reveals protein–protein interactions enabling iron release from bacterioferritin. J Am Chem Soc 134(32):13470–13481
8. Tosha T, Ng H-L, Bhattasali O, Alber T, Theil EC (2010) Moving metal ions through ferritin – protein nanocages from three-fold pores to catalytic sites. J Am Chem Soc 132:14562–14569
9. Mendel D, Ellman JA, Chang Z, Veenstra DL, Kollman PA, Schultz PG (1992) Probing protein stability with unnatural amino acids. Science 256:1798–1802
10. Karpusas M, Baase WA, Matsumura M, Matthews BW (1989) Hydrophobic packing in T4 lysozyme probed by cavity-filling mutants. Proc Natl Acad Sci U S A 86:8237–8241
11. Saito M, Kono H, Morii H, Uedaira H, Tahirov TH, Ogata K, Sarai A (2000) Cavity-filling mutations enhance protein stability by lowering the free energy of native state. J Phys Chem B 104:3705–3711
12. Ishikawa K, Nakamura H, Morikawa K, Kanaya S (1993) Stabilization of *Escherichia coli* ribonuclease HI by cavity-filling mutations within a hydrophobic core. Biochemistry 32:6171–6178
13. Eijsink VGH, Dijkstra BW, Vriend G, van der Zee JR, Vettman OR, van der Vinne B, van den Burg B, Kempe S, Venema G (1992) The effect of cavity-filling mutations on the thermostability of *Bacillus stearothermophilus* neutral protease. Protein Eng 5:421–426
14. Akasako A, Haruki M, Oobatake M, Kanaya S (1997) Conformational stabilities of *Escherichia coli* RNase HI variants with a series of amino acid substitutions at a cavity within the hydrophobic core. J Biol Chem 272:18686–18693
15. Kono H, Saito M, Sarai A (2000) Stability analysis for the cavity-filling mutations of the Myb DNA-binding domain utilizing free-energy calculations. Proteins 38:197–209
16. Ohmura T, Ueda T, Ootsuka K, Saito M, Imoto T (2001) Stabilization of hen egg white lysozyme by a cavity-filling mutation. Protein Sci 10:313–320
17. Tanaka M, Chon H, Angkawidjaja C, Koga Y, Takano K, Kanaya S (2010) Protein core adaptability: crystal structures of the cavity-filling variants of *Escherichia coli* RNase HI. Protein Pept Lett 17:1163–1169
18. Shortle D, Stites WE, Meeker AK (1990) Contributions of the large hydrophobic amino acids to the stability of staphylococcal nuclease. Biochemistry 29:8033–8041
19. Ardejani MS, Chok XL, Foo CJ, Orner BP (2013) Complete shift of ferritin oligomerization toward nanocage assembly via engineered protein-protein interactions. Chem Commun 49:3528–3530
20. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE (2000) The Protein Data Bank. Nucleic Acids Res 28:235–242
21. Eric FP, Thomas DG, Conrad CH, Gregory SC, Daniel MG, Elaine CM, Thomas EF (2004) UCSF chimera - a visualization system for exploratory research and analysis. J Comput Chem 25:1605–1612
22. Schymkowitz J, Borg J, Stricher F, Nys R, Rousseau F, Serrano L (2005) The FoldX web server: an online force field. Nucleic Acids Res 33:W382–W388
23. Binkowski TA, Naghibzadeh S, Liang J (2003) CASTp: Computed Atlas of Surface Topography of proteins. Nucleic Acids Res 31:3352–3355
24. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990) Basic local alignment search tool. J Mol Biol 215:403–410
25. Larkin MA, Blackshields G, Brown NP, Chenna R, McGettigan PA, McWilliam H, Valentin F, Wallace IM, Wilm A, Lopez R, Thompson JD, Gibson TJ, Higgins DG (2007) Clustal W and Clustal X version 2.0. Bioinformatics 23:2947–2948
26. O'Neil KT, DeGrado WF (1990) How calmodulin binds its targets: sequence independent recognition of amphiphilic [alpha]-helices. Trends Biochem Sci 15:59–64
27. Gellman SH (2002) On the role of methionine residues in the sequence-independent recognition of nonpolar protein surfaces. Biochemistry 30:6633–6636
28. Hu D, Qin Z, Xue B, Fink AL, Uversky VN (2008) Effect of methionine oxidation on the structural properties, conformational stability, and aggregation of immunoglobulin light chain LEN. Biochemistry 47:8665–8677