A machine learning based approach to side chain optimization

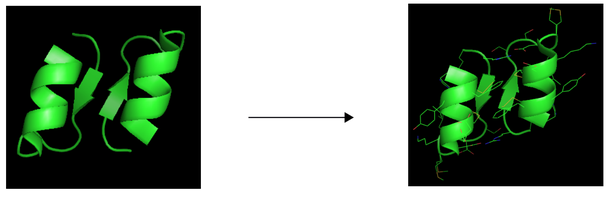
Sabareesh Subramaniam[1] , Sriraam Natarajan[2] and Alessandro Senes [1]

[1]Dept of Biochemistry, University of Wisconsin-Madison and [2]Translational Sciences Institute, Wake Forest University, Winston-Salem, NC

Abstract

Side chain optimization refers to the problem of repacking sidechain atoms on a fixed backbone so as to minimize the energy of the resultant structure. It is typically performed as a search over the combinatorial space of conformations for all the positions in the backbone. The finite set of representative conformations sampled for each amino acid type is called a ``conformer library''. Optimization procedures do not take into account the fact that each position in a protein backbone has different sampling (number of conformations) needs, for example, solvent exposed positions require less sampling than positions buried in the core of the protein. The key contribution of this work is a method to distribute conformations among different positions in a protein backbone based on their sampling needs using machine learning. Our results demonstrate that this strategy helps to redistribute sampling efficiently and helps achieve lower energies.

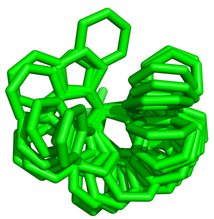
Sidechain Optimization



A side chain optimization problem is defined by a template (backbone) and a set of energy functions, which define a continuum energy lanscape covering all possible theoretical side chain conformations.

The goal of sidechain optimization is to repack the sidechain conformations onto the backbone so as to achieve the global minimum energy conformation.

Using a conformer library leads to a combinatorial search space



At each position in the backbone, the sidechain is allowed to assume any one of a finite number of conformations. This set of conformations is called a conformer/rotamer library.

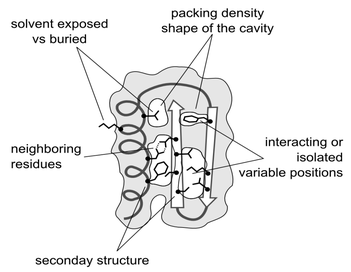
The conformer library is typically constructed using statistics from existing protein structures or by applying some geometrical criteria on natural sidechain conformations.

Sidechain optimization thus reduces to the problem of deciding the conformation at each position in the backbone from among the conformations in the library.

 # possible conformations = 10 * 3 * 10 * 3 * 10 * 9 = 81000

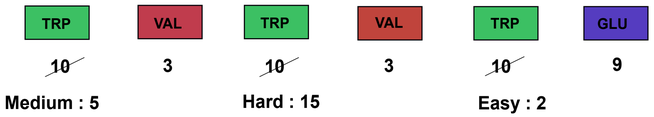
The number of possible conformations grows combinatorially with the number of conformers at each position. Typically, the same number of conformers are assigned to all positions containing a particular amino acid sidechain. In the figure above, all TRPs have 10 conformers, all VALs have 3 conformers and all GLUs have 9 conformers.

The combinatorial complexity may be reduced by predicting the sampling requirement for each side chain based on its immediate environment.



To achieve good energy, a buried position may need a very specific conformation whereas an exposed position may have multiple isoenergetic solutions. Thus, every position has different sampling requirements.

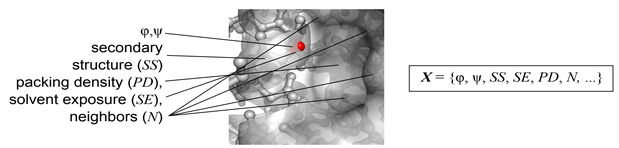
If we can predict the sampling requirement of each position, we can then re-allocate sampling optimally, reducing the combinatorial complexity and/or achieving better energy.



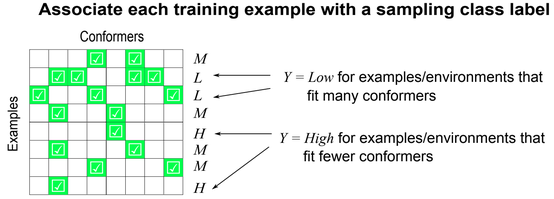
 # possible conformations = 5 * 3 * 15 * 3 * 2* 9 = 12150

The sampling requirements of each position in a sidechain optimization problem may be predicted using machine learning.

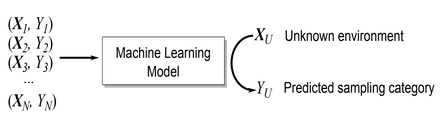
Associate each training example (sidechain) with a feature vector X



Associate each training example with a sampling class label Y

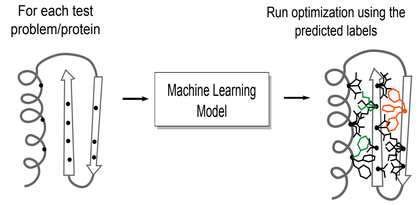


Train the Learner with the (X, Y) pairs

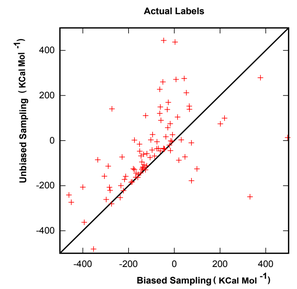


Using the labels predicted by the machine learning algorithms helps achieve lower energies than an equivalent unbiased method.

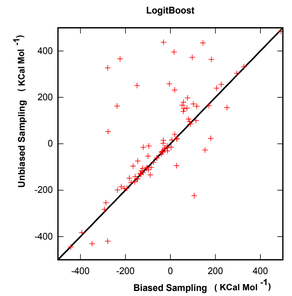
The following classification algorithms, as implemented in WEKA, were used 1) Bagging 2) CART 3) LogitBoost 4) NaiveBayes



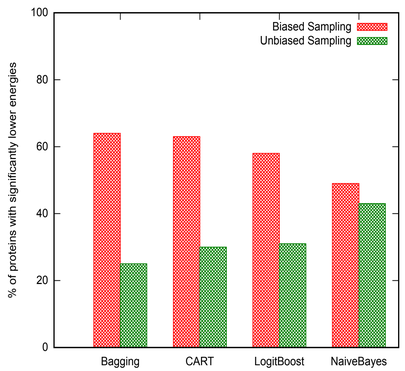
The labels predicted by all the classifiers were used to perform sidechain optimization and the energies obtained were compared against an unbiased sampling of comparable size. All the classifiers used were able to achieve lower energies than the unbiased scheme in a majority of the test set proteins



a) Comparison of energies achieved using the actual labels against the energies achieved using an equivalent unbiased sampling scheme. The biased sampling scheme achieved lower energies than the unbiased scheme in a majority of the proteins.



b) Comparison of energies achieved using the labels produced by LogitBoost against the energies achieved using an equivalent unbiased sampling scheme. The biased sampling scheme achieved lower energies than the unbiased scheme in a majority of the proteins.



c) The bar graph shows a comparison of each classifier against its equivalent unbiased sampling scheme. All algorithms except Naive Bayes perform very well compared to the unbiased sampling scheme.

It is important to note that we have allowed the number of conformers in the unbiased scheme to vary for each protein so that the search space sizes are comparable. This is not the case in typical sidechain optimization procedures where typically the same number of conformers are used for all proteins. Therefore, even in such disadvantaged circumstances, the biased classification scheme proves to be more efficient than the unbiased scheme.

References

1. Dunbrack RL, Jr. and Cohen FE. (1997) *Protein Science* **6**, 1661-1681
2. Xiang Z, Honig B. (2001) *J Mol Biol* **311**, 421-30
3. Shetty RP, De Bakker PIW, DePristo MA, Blundell TL. (2003) *Protein Eng* **16** 963-9.
4. Mark Hall et al., (2009) The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1.