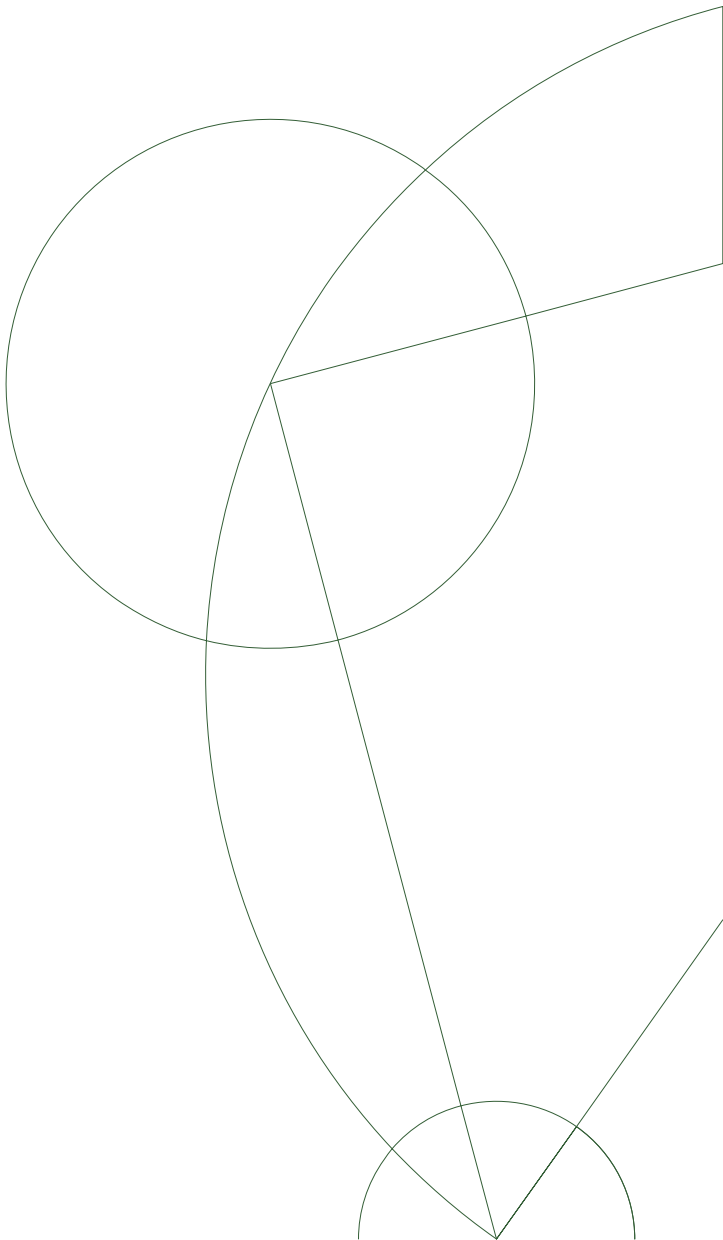
FACULTY OF SCIENCE
UNIVERSITY OF COPENHAGEN

PhD Thesis

Anders S. Christensen

Protein Structure Determination Using Chemical Shifts



Academic supervisor: Jan H. Jensen

April 24, 2014

Acknowledgement

This thesis represents the work I have carried out as a PhD student under the supervision of Professor Jan H. Jensen in his group of Biocomputational Chemistry. Thank you to all who have supported me during my work at the third floor of C-building at the H.C. Ørsteds Institute.

I would especially like to thank the following people:

- Thank you to my supervisor, Jan "Yoda" Jensen for introducing me to the exciting fields of quantum chemistry and biocomputational chemistry, teaching me everything I know (and more), for your patience, and the inspiration you bring to everyone around you.
- Thank you to Jens Breinholt at Novo Nordisk for supporting me with my work – I sincerely hope, that my work will very soon become practically useable. And thank you to the Novo Nordisk STAR PhD program for financial support, and for giving me the opportunity to carry out this study.
- Thank you to all of our collaborators at the Biocenter, who always have been very supportive. Especially, Thomas Hamelryck (in the presence of whom everything is trivially solved using Bayes' theorm) for helping me out with Bayesian theory, your great ideas and more, Wouter Boomsma for being seemingly all-knowing in what concerns PHAISTOS and always being exceptionally helpful, Simon Olsson for helping out with the implementation of the Jeffrey's prior code, and Kresten Lindorff-Larsen for always being encouraging and sharing your knowledge in this field.
- Thank you to my office mates, Casper Steinmann, Jimmy Kromann and Lars Bratholm, for the invaluable company, our office pranks, and endless number of energy drinks consumed, as well as the highly valuable scientific discussions we continue to share daily (not forgetting the virtual monster we've slayed).

  Also thank you to those close colleagues who came by for coffee and friendly conversations; Jonas Elm, Jacob Lykkebo, Nini Reeler, Frederik Beyer (and many, many more!).
- Thank you to everyone at the Department of Chemistry, especially Kurt V. Mikkelsen, Stephan P. A. Sauer and Sten Rettrup for always being so helpful with everything from bureaucratic procedures, to coupled cluster theory, to derivation of the Slater-Kloster tables.
- Thank you to all the students in the courses I've taught, and especially the very talented students who have carried out Master's, Bachelor's and various research projects under my supervision. Of those not already mentioned (and in no particular order): Maher Channir, Anders Larsen, Rie Nielsen, Christine Skibsted, Cecilie Lindholm.
- Thank you to everyone I forgot to mention, including all the unnamed developers of the free, open source software I use in my daily work – the Open Babel project in particular.

Lastly, an even bigger thanks goes to my IRL family and friends, whom I have been seeing much less than I should since I undertook my PhD studies. Thanks, everyone!

Dansk Resumé

Kemien af et protein er tæt forbundet med dens tre-dimensionelle struktur. Af denne grund, er proteinstruktur bestemmelse grundlaget for rationel forståelse af kemien af biologiske processer, der involverer proteiner.

For tiden er flest kendte proteinstrukturer blevet løst ved røntgenkrystallografi. Kravet til løsning af en struktur på denne måde er, at proteinet krystalliserer. Moderne krystaliseringsmetoder dog kun har en succesrate på 5% [Warke and Momany, 2007]. I disse tilfælde kan kernemagnetisk resonans (NMR) metoder anvendes med en vis succes. I øjeblikket indeholder Protein Data Bank 90.000 strukturer løst ved røntgen- og 9.000 strukturer løst ved NMR-metoder, og omkring 10.000 røntgen- og 500 NMR-strukturer bliver indsendt hvert år [Berman et al., 2000].

Konventionelle NMR-metoder til bestemmelse af protein strukturer optager et flerdimensionelt spektrum, som korrelerer resonansfrekvenser flere kerner på samme tid. Fra dette spektrum er først problem at tilordne de kemiske skift af hver kerne. Denne proces er i vid udstrækning automatiseret for hovedkædeatomer, men er mere involveret for sidekædeatomer. Disse oplysninger bruges til at identificere toppe i spektret, der svarer til afstandsbegrænsninger (NOE begrænsninger) mellem par af atomer. Disse distance begrænsninger er det bruges til at generere ensembler af strukturer, der tilfredsstiller det givne sæt af begrænsninger. Protein NMR-spektroskopi har imidlertid flere begrænsninger. Store proteiner har meget overfyldte spektre , hvilket komplicerer opgaven - hovedsagelig på grund af brede toppe og resulterende spektraloverlapning. Dette er en væsentlig hindring for tilordningen af de kemiske skift og dermed for at finde de værdifuld NOE begrænsninger. Følgeligt har omkring 95 % af alle NMR- strukturer i PDB-databasen således har en størrelse på kun 200 aminosyrer eller mindre. Dette kan sammenlignes med de gennemsnitlige størrelser af proteiner i mennesker og *E. coli*, som er henholdsvis omkring 400-600 og 200-400. Problemet kan mindskes ved deuterering som imidlertid falder til nummer NOE-begrænsninger, der kan findes. Isotopmærkningsmetoder som selektivt mærker visse sidekæder er blevet udviklet som en effektiv strategi for sådanne problemer.

Computerberegningsmetoder

En anden tilgang til at løse en proteinstruktur fra aminosyresekvensen er simulering af energilandskabet af proteinet. Dette kaldes også proteinfoldning. I denne tilgang, er de mulige konformationer samplet og scoret med en beskrivelse af proteinernes fysik, uden ekstra viden fra eksperimenter. Sådanne *ab initio* tilgange har været anvendet til at bestemme strukturer, typisk med en præcision ned til 3 Å, via Monte Carlo simuleringer i ROSETTA-programmet [Rohl et al., 2004]. Et andet næveværdigt eksempel er den samtidige bestemmelse af struktur og dynamik flere små proteiner via meget lange molekylær dynamik (MD) simuleringer med Anton computer [Lindorff-Larsen et al., 2005].

Selv om disse metoder ikke kræver noget eksperimentelt arbejde, er det ekstremt krævende i forhold til de edb-ressourcer, der er nødvendige. Desuden er de normalt ikke nemme at konvergere for systemer > 100 aminosyrer [Lange and Baker, 2012]. ROSETTA-metoden er (i øjeblikket) velsagtens den mest succesfulde metode til at bestemme en proteinstruktur via computer beregninger. For nylig viste Baker gruppen, at optagelsen af hovedkæde kemiske skift og RDC

data forbedrer ROSETTA-protokollen og tillader bestemmelse af strukturer op til 150 rester [Raman et al., 2010, Lange and Baker, 2012].

Grundlaget for ROSETTA er fragment-samling af lokale proteinstrukturmodeller, kombineret med raffinering ved hjælp af en energifunktion, der er blevet påvist at fungere bemærkelsesværdigt godt. Kort beskrevet består fuldatom-ROSETTA-energifunktion af flere additive temer som Lennard-Jones potentialer, termer for eksponering solvent, hydrogenbindinger, elektrostatiske par-interaktioner og dispersion-iteraktioner, og endelig torsions potentialer for hovedkæde- og sidekædevinkler.

Nøjagtigheden af energifunktionen kommer dog på bekostning af beregningsmæssige hastighed og ufuldstændig i den konformationelle prøvetagning, som synes at være den uoverkommelige forhinding for yderligere succes for ROSETTA. Denne protokol er for nylig blevet forbedret yderligere med inddragelse af meget sparsomme mængder NOE-data [Lange et al., 2012].

Dette gav 7 strukturer omkring 200 aminosyrer, der blev bestemt med en nøjagtighed på mellem 2,5 og 3,9 Å fra de tilsvarende eksperimentelle røntgen-strukturer, og desuden blev en god struktur for det 376 aminosyrer store maltosebindingsprotein endda fundet, men dette krævede væsentligt flere NOE oplysninger. Disse simuleringer krævede en 512-kerner supercomputer som kørte i flere dage, for hvert protein.

Et andet nævneværdigt eksempel på protein strukturbestemmelse metoder, der beskæftiger NMR-data, er er CHESHIRE-metoden [Cavalli et al., 2007]. CHESHIRE-metoden var den første metode som løste strukturer kun ved brug af kemiske skift, og bruger en fragmentsamlingstilgang, efterfulgt af en Monte Carlo raffinering ved hjælp af et all-atom kraft-felt og en energi-funktion, der inkluderer kemiske skift. Denne metode blev anvendt til at bestemme proteinstrukturer fra kemiske skift, og fandt strukturer for 11 proteiner mellem 54 og 123 aminosyrer i størrelse, til en nøjagtighed på omkring 1,5 Å fra de tilsvarende eksperimentelle røntgen-strukturer.

I det følgende afsnit, er PHAISTOS-programmet introduceret, og formalisme for inkludering af kemiske skift i simuleringer i PHAISTOS er udledt. Dette er et forsøg på at løse to centrale udfordringer i proteinfoldning: (1) Fuldstændig konformationel prøveudtagning og (2) nøjagtig energi-scoring af konformationelle prøver. Disse udfordringer er mødt som følger: (1) ved hjælp af en nyudviklet forudindtaget konformationel prøveudtagningsmetode og (2) ved at parametrisere en nøjagtig kemisk skift forudsigelsesmetode, brut med en energifunktion baseret på Bayesiansk statistik, som tillader, at dette kombineres med eksisterende energifunktioner i PHAISTOS. Denne kombinerede fremgangsmåde vil blive demonstreret på foldningssimuleringer på et testsæt af proteiner med kendte strukturer spænder fra 55 til 269 rester.

Publication list

List of publications:

1. Anders S. Christensen, Stephan P. A. Sauer, Jan H. Jensen (2011) Definitive benchmark study of ring current effects on amide proton chemical shifts. *Journal of Chemical Theory and Computation*, 7:2078-2084.

2. Wouter Boomsma, Jes Frellsen, Tim Harder, Sandro Bottaro, Kristoffer E. Johansson, Pengfei Tian, Kasper Stovgaard, Christian Andreetta, Simon Olsson, Jan B. Valentin, Lubomir D. Antonov, Anders S. Christensen, Mikael Borg, Jan H. Jensen, Kresten Lindorff-Larsen, Jesper Ferkinghoff-Borg, Thomas Hamelryck (2013) PHAISTOS: A framework for Markov chain Monte Carlo simulation and inference of protein structure. *Journal of Computational Chemistry*, 34:1697-1705.

3. Anders S. Christensen, Troels E. Linnet, Mikael Borg, Wouter Boomsma, Kresten Lindorff-Larsen, Thomas Hamelryck, Jan H. Jensen (2013) Protein Structure Validation and Refinement Using Amide Proton Chemical Shifts Derived from Quantum Mechanics. *PLoS ONE* 8:e84123.

4. Anders S. Christensen, Thomas Hamelryck, Jan H. Jensen (2014) FragBuilder: An efficient Python library to setup quantum chemistry calculations on peptides models. *PeerJ* 2:e277.

List of public code:

1. FragBuilder (BSD license) `https://github.com/jensengroup/fragbuilder/`
2. CamShift module (BSD license) `https://github.com/jensengroup/camshift-phaistos/`
3. ProCS module (BSD license) `https://github.com/jensengroup/procs-phaistos/`
4. PHAISTOS (GPL license) `http://sourceforge.net/projects/phaistos/`
5. GAMESS patch FMO-RHF:MP2 (GAMESS license/free) `https://github.com/andersx/fmo-rhf-mp2/`
6. PHAISTOS GUI (BSD license) `https://github.com/andersx/guistos/`
7. NOE module (BSD license) `https://github.com/andersx/noe-way-jose/`

List of other publications:

1. Casper Steinmann, Kristoffer L. Blædel, Anders S. Christensen, Jan H. Jensen (2013) Interface of the polarizable continuum model of solvation with semi-empirical methods in the GAMESS program. *PLoS ONE* 8:e67725.

2. Anders S. Christensen, Casper Steinmann, Dmitri G. Fedorov, Jan H. Jensen (2013) Hybrid RHF/MP2 geometry optimizations with the Effective Fragment Molecular Orbital Method. *PLoS ONE* 9:e88800

3. Jimmy C. Kromann, Anders S. Christensen, Casper Steinmann, Martin Korth, Jan H. Jensen (2014) A third-generation dispersion and third-generation hydrogen bonding corrected PM6 method: PM6-D3H+. *PeerJ PrePrints* 2:e353v1.

Contents

Chapter 1

Introduction

The chemistry of a protein is tightly linked to its 3-dimensional structure. For this reason, protein structure determination is the basis of rational understanding of the chemistry of biological processes involving proteins.

Most currently known protein structures have been solved by X-ray crystallography. One requirement for solving a structure this way is that the protein will crystallize. Modern crystallization methods, however, only have a success rate of 5% [Warke and Momany, 2007]. In these cases, nucleic magnetic resonance (NMR) methods may be used with some success. Currently the Protein Data Bank contain 90,000 structures solved by X-ray and 9,000 structures solved by NMR methods, and around 10,000 X-ray and 500 NMR structures are being submitted each year [Berman et al., 2000].

Conventional NMR protein structure determination methods records a multidimensional spectrum that correlate the resonance frequencies of several nuclei at the same time. From this spectrum, the common work flow is to first assign the chemical shifts of each nuclei. This process is largely automated for backbone nuclei, but is more involved for side chain atoms. This assignment information is used to identify peaks in the spectrum which correspond to distance restraints (NOE restraints) between pairs of atoms. These distance restraints are the used to generate ensembles of structures that satisfy the given set of restraints.

Protein NMR spectroscopy, however, has several limitations. Large proteins have very crowded spectra, which complicates assignment, mostly due to broad peaks and resulting spectral overlap. This is a substantial hindrance to assignment of the chemical shifts, and therefore obtaining the valuable NOE restraints. Consequently, around 95% of all NMR structures in the PDB database thus have a size of only 200 amino acids or less. This can be compared to the average sizes of proteins in humans and *E. coli*, which are around 400-600 and 200-400, respectively. These problem can be somewhat alleviated by deuteration which, however, decreases to number NOE restraints that can be obtained. Isotope labeling schemes which selectively label only certain side chains have been invented, as an efficient strategy for such problems.

1.1 Computational methods

A different approach to solving a protein structure from the amino acid sequence is simulation of the energy landscape of the protein. This practice is also referred to as *protein folding*. In this approach, the possible conformations are sampled and scored using a description of the physics of the proteins, with no extra knowledge from experiments. Such *ab initio* approaches have been used to determine structures up to an accuracy of typically 3 Å using Monte Carlo simulations the ROSETTA program [Rohl et al., 2004]. Another notable example is the simultaneous determination of structure and dynamics of several small proteins via very long molecular dynamics (MD) simulations using the Anton computer [Lindorff-Larsen et al., 2005].

While these methods do not require any experimental input, the are extremely demanding in terms of the computational resources that are required. Furthermore, they usually fail to converge for structures > 100 amino acids [Lange and Baker, 2012].

The ROSETTA methodology is (currently) arguably the most successful method to determine a protein structure computationally. Recently, the Baker group showed, that inclusion of backbone chemical shifts and RDC data vastly improved the ROSETTA protocol and allowed structures up to 150 residues to be determined [Raman et al., 2010, Lange and Baker, 2012]. The basis of ROSETTA fragment-assembly of local protein structure, combined with refinement using an energy function that has been demonstrated to work remarkably well. Briefly described, the all-atom ROSETTA energy function consists of several additive terms such as Lennard-Jones potentials, terms for solvent exposure, hydrogen bonding, electrostatic pair-interactions and dispersion interactions, and finally torsional potentials for backbone and side chain angles. The demonstrated accuracy of the energy function does come at the cost of computational speed and incomplete conformational sampling seems to be the prohibitive for further success for ROSETTA. This protocol has recently been further improved with inclusion of very sparse NOE data [Lange et al., 2012].

This allowed 7 structures around 200 amino acids to be determined, to an accuracy of between 2.5 and 3.9 Å from the corresponding experimental X-ray structures Furthermore, a good structure for the 376 amino acids maltose binding protein could even be determined, but this required substantially more NOE data. These simulations, however required a 512-cores super computer for running several days, for each protein.

Another notable example of protein structure determination methods that employ NMR data is is the CHESHIRE method [Cavalli et al., 2007]. The CHESHIRE method was the first method which solved structures using only chemical shifts, and uses a fragment-assembly approach followed by a Monte Carlo refinement using an all-atom force-field and an energy function that includes chemical shifts. This method was used to determine the protein structures from chemical shifts, and was demonstrated on 11 proteins between 54 and 123 amino acids in size to an accuracy of around 1.5 Å from the corresponding experimental X-ray structures.

In the following section, the PHAISTOS program is introduced, and the formalism for inclusion of chemical shifts in simulations in PHAISTOS is derived. This is an attempt to address the two central challenges in protein folding: (1) complete conformational sampling and (2) accurate energy scoring of conformational samples.

These challenges are met as follows: (1) using a recently developed biased conformational sampling method and (2) by parametrizing an accurate chemical shift predictor and deriving an energy function based rigorously on Bayesian statistics, which allows this to be combined with existing energy functions in PHAISTOS.

The combined approach will be demonstrated on folding simulations on a test-set of protein with known structures ranging from 55 to 269 residues.

Chapter 2

Introduction to PHAISTOS

This section servers as an introduction to the PHAISTOS program, and a (very) brief introduction to the theory behind PHAISTOS [Boomsma et al., 2013]. This will give the relevant background to read the next chapters. PHAISTOS is also published and discussed in detail in paper #2 in this the appendix.

2.1 Markov Chain Monte Carlo

One of the primary goals of simulations in PHAISTOS is to construct the Boltzmann distribution of a protein via Markov chain Monte Carlo (MCMC) sampling for a given potential energy surface at a given temperature. The Boltzmann distribution of a protein structure, $\mathbf{X}$, at a given temperature, T, is given by:

$$p(\mathbf{X}) = \frac{1}{Z(T)} \exp\left(\frac{-E}{k_{\mathrm{B}}T}\right), \tag{2.1}$$

where $k_{\mathrm{B}}T$ is Boltzmann's constant and $Z(T)$ is the partition function at the given temperature.

In Markov chain Monte Carlo the target distribution obtained by repeatedly proposing updates to the current state, and accepting or rejecting these updates with a certain acceptance probability.

It can be shown, that for an infinitely sampled distribution to converge to the correct target distribution, i.e. $p_\infty(\mathbf{X}) = p(\mathbf{X})$, the Monte Carlo moves that are used to propose updates must satisfy the principle of detailed balance. That is, the transition from the current state $\mathbf{X}$ to the proposed new state $\mathbf{X}'$ fulfills:

$$p(\mathbf{X})p(\mathbf{X} \to \mathbf{X}') = p(\mathbf{X}')p(\mathbf{X}' \to \mathbf{X}) \tag{2.2}$$

where $p(\mathbf{X} \to \mathbf{X}')$ is the probability to of moving from the state $\mathbf{X}$ to $\mathbf{X}'$ using a given move. If we further factorize $p(\mathbf{X} \to \mathbf{X}')$ into an acceptance probability p_a and a move transition probability p_m, Eqn. 2.2 gives:

$$\frac{p_a(\mathbf{X} \to \mathbf{X}')}{p_a(\mathbf{X}' \to \mathbf{X})} = \frac{p(\mathbf{X}')}{p(\mathbf{X})} \frac{p_m(\mathbf{X}' \to \mathbf{X})}{p_m(\mathbf{X} \to \mathbf{X}')} \tag{2.3}$$

Most of the moves in PHAISTOS are symmetric, that is the move bias ratio $p_m(\mathbf{X}' \to \mathbf{X})/p_m(\mathbf{X} \to \mathbf{X}') = 1$, but for some moves this is not true. These biased moves can be exploited to vastly speed up convergence or bias the simulation, and are discussed later in Section 2.2.

2.1.1 Metropolis-Hastings

The simplest Monte Carlo method that satisfies Eqn. 2.3 is the Metropolis-Hastings method. Here a transition $\mathbf{X} \to \mathbf{X}'$ is accepted using the Metropolis-Hastings acceptance criterion:

$$p_a(\mathbf{X} \to \mathbf{X}') = \min\left(1, \frac{p(\mathbf{X}')}{p(\mathbf{X})} \frac{p_m(\mathbf{X}' \to \mathbf{X})}{p_m(\mathbf{X} \to \mathbf{X}')}\right) \tag{2.4}$$

Evaluation of the partition function is thus not necessary. The Metropolis-Hastings method is efficient when exploring native states, and simulations near the critical temperature. Unfortunately the Metropolis-Hastings method, compared to other MC methods, often gets stuck in local minima, and is therefore generally inefficient when simulating protein folding from an extended strand.

2.1.2 Generalized Ensembles

To avoid the slow convergence problem advanced MC methods are available in PHAISTOS, which emphasize sampling at low energies, which is generally of higher interest in protein structure determination. These "generalized ensemble" methods are very similar to the Metropolis-Hastings method, and the main difference in the acceptance criterion is that the target distribution $p(\mathbf{X})$ has been replace by a generalized weight function $w(\mathbf{X})$. The acceptance criterion then becomes:

$$p_a(\mathbf{X} \to \mathbf{X}') = \min\left(1, \frac{w(\mathbf{X}')}{w(\mathbf{X})} \frac{p_m(\mathbf{X}' \to \mathbf{X})}{p_m(\mathbf{X} \to \mathbf{X}')}\right) \tag{2.5}$$

Through reweighting, samples from a converged simulation in a generalized ensemble can be reweighted to correspond to the Boltzmann distribution at a given temperature.

PHAISTOS offer two generalized ensemble methods. In the multicanonical ensemble method, the weight function is $w_{\text{muca}}(\mathbf{X}) = 1/g(E(\mathbf{X}))$, where $E(\mathbf{X})$ is the energy of the structure $\mathbf{X}$ and g is the associated density of states. In the inverse-k ensemble, the weight function is given by $w_{1/k}(\mathbf{X}) = 1/k(E(\mathbf{X}))$ where $k(E(\mathbf{X})) = \int_{-\infty}^{E(\mathbf{X})} g(E')dE'$. The since the density of states is generally unknown, the weight-function is estimated during the simulation. PHAISTOS uses the MUNINN library to collect histograms of the energy and efficiently provide an estimate of $w(\mathbf{X})$ on-the-fly [Ferkinghoff-Borg, 2002].

2.2 Monte Carlo Moves Using Generative Probabilistic Models

PHAISTOS proposes new structure samples using a weighted set of difference MC moves, which each randomly changes the current protein structure in a certain way. Briefly, these are divided in side chain moves and backbone moves. Side chain moves update the rotamer-conformation of a amino-acid single side chain by rotating the dihedral angles on the side chain. Backbone moves either perform a local perturbation to a strand of a only a few amino acids, or rotates one dihedral angle on the backbone.

Using random moves which re-sample angles from a uniform distribution, and then constructing a target distribution via an acceptance criterion is a perfectly valid strategy. However, sampling from a uniform distribution usually lead to slow convergence. A common approach to alleviate this problem is using fragment assembly, in which small fragments of peptides are assembled from a library of common fragment motifs, such as beta-strands, helices and loops. This approach, however, introduces a move bias, which must be divided out if the simulation has to obey detailed balance. Furthermore, it is not clear, how to evaluate the move bias ratio $p_m(\mathbf{X}' \to \mathbf{X})/p_m(\mathbf{X} \to \mathbf{X}')$ when sampling from a fragment library.

A related approach to obtain a similar speed up is biased sampling. PHAISTOS supports sampling of both side chain and backbone angles from such generative probabilistic models. In this approach, angles are sampled from distributions that are conditioned on prior knowledge. Two all-atom generative probabilistic models are supported in PHAISTOS. TorusDBN which is a hidden-Markov model of backbone angles [Boomsma et al., 2008], and BASILISK [Harder et al., 2010] which is a similar model of side chain rotamer-conformations. Both work are continuous models in torsion-angle space. The model that is used in this work is TorusDBN, which is is a model that samples backbone dihedral angles conditioned on the amino acid sequence from a distribution that resembles the Ramachandran-plot. This effectively speeds up convergence of sampling, since uninteresting parts of conformational space in only sampled very rarely. The importance of the TorusDBN model is discussed in chapter 6.

Using models such as TorusDBN and BASILISK introduces a move bias, which compensated for in Eqn. 2.3 by multiplying by the ratio $p_m(\mathbf{X}' \to \mathbf{X})/p_m(\mathbf{X} \to \mathbf{X}')$. It is possible to determine this ratio, because the likelihood of sampled values can be calculated in the TorusDBN model. It is thus possible to recover the target distribution (e.g. the Boltzmann distribution or a generalized ensemble), despite using only biased moves.

Effectively, this turns the target distribution into an effective target distribution. For sampling from the Boltzmann distribution (e.g. using a molecular mechanics force field), the effective target distribution becomes

$$p_e(\mathbf{X}) = p(\mathbf{X})p_m(\mathbf{X}|I), \tag{2.6}$$

where $p_m(\mathbf{X}|I)$ is the probability distribution from the generative model, conditioned on the prior information I available to the model. This is approach is formally equivalent to adding the term $\ln(p_m(\mathbf{X}|I))$ to the physical energy (although this term does not scale with the temperature):

$$\begin{aligned} p_e(\mathbf{X}) &= p(\mathbf{X})p_m(\mathbf{X}|I) \\ &\propto \exp\left(\frac{-E(\mathbf{X})}{k_\mathrm{B}T}\right)p_m(\mathbf{X}|I) \\ &\propto \exp\left(\frac{-E(\mathbf{X})}{k_\mathrm{B}T} - \ln\left(p_m(\mathbf{X}|I)\right)\right) \end{aligned} \tag{2.7}$$

In other words, biased sampling can be regarded as simply use of a better force field, while the convergence of the simulation is vastly improved.

TorusDBN is implemented in two versions; standard TorusDBN which, in brief, is conditioned on only the amino-acid sequence, and TorusDBN-CS which is furthermore based on backbone and

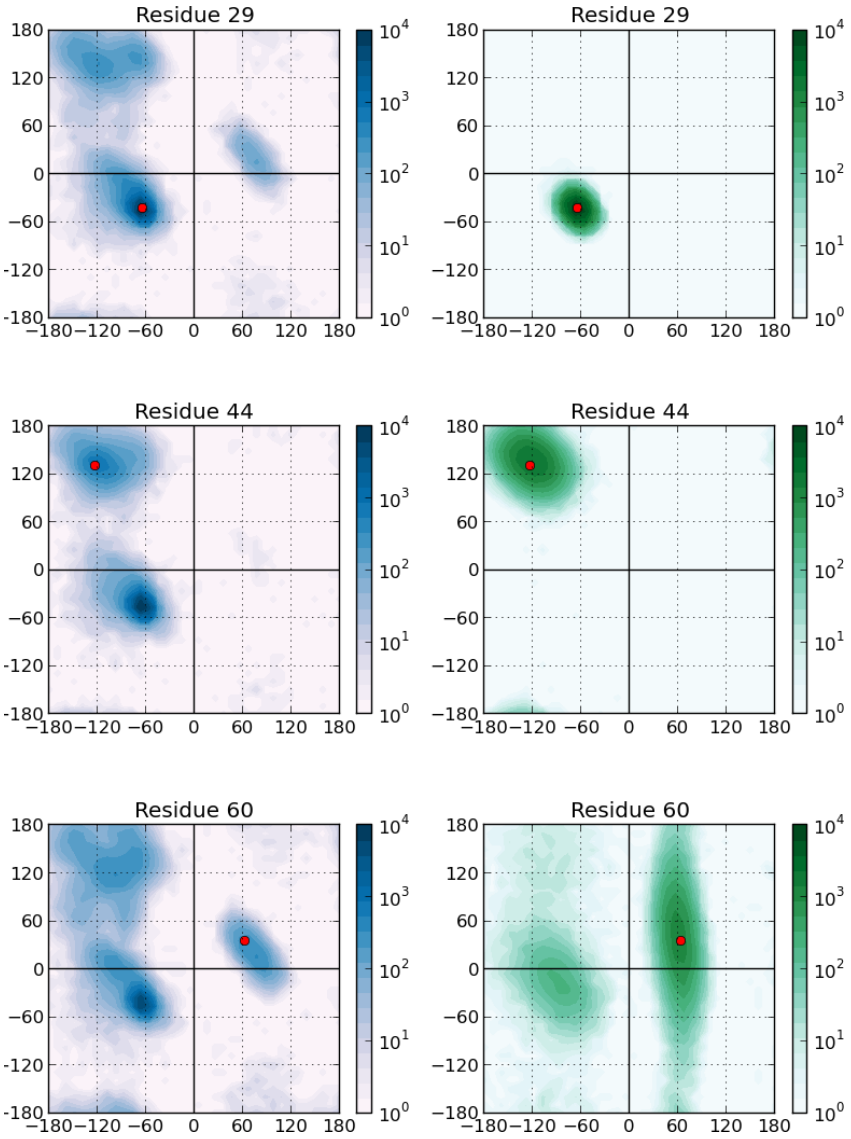


Figure 2.1: Sampling densities from TorusDBN (left/blue) and TorusDBN-CS (right/green) for the residues 29, 44 and 60 in Ubiquitin. Values from the experimental structure 1UBQ are marked with a red dot. Residue 29 (lysine) is located in the middle of an alpha-helix. Residue 44 (isoleucine) is located in a beta-sheet motif, and finally residue 60 (aspargine) is located in a loop region.

beta-carbon chemical shifts. The default TorusDBN model is trained on a set of 1,447 proteins of 180 different SCOP-fold classifications. The default TorusDBN-CS model is trained on 1349 proteins and corresponding chemical shifts from the RefDB training set.

Effectively, proposing structures from TorusDBN biases the simulation towards likely angles within the Ramachandran-plot, and furthermore also towards a certain secondary structure type that is likely for the particular amino acid sequence. The effect of TorusDBN-CS is similar, but the effect is much more pronounced.

Fig. 2.1 shows an example of three different, but typical cases from Ubiquitin. These are alpha-helix, beta-sheet and loop regions. Residue 29 (lysine) is in a typical alpha helix and this corresponds to the most often sampled cluster from both TorusDBN and TorusDBN-CS. TorusDBN-CS, however, very precisely locates the center of the cluster to within around ± 15 degrees. TorusDBN, in contrast, has some sampling density in the regions typical for beta-sheet and left-handed alpha-helices.

For residue 44 (isoleucine) which is in a typical alpha-helix region of Ubiquitin, TorusDBN-CS accurately pinpoints the distribution of samples around the experimental values. TorusDBN, however, manages to rule out left-handed helices, but has a higher sampling density in the alpha-helix region than the beta-sheet region.

The last residue in the examples, residue 60 (aspargine), is located in a loop-region with backbone angles that correspond to a left-handed helix. Both models sample in the correct region, but TorusDBN favor a regular alpha-helix. While TorusDBN-CS heavily favors the correct region, angles that are usually not favored in the Ramachandran plot are also frequently sampled in this particular case. This is presumably due to less fold-diversity in the training set, compared to the set used to train TorusDBN. Generally, however, the TorusDBN-CS distribution is more restrictive than standard TorusDBN.

2.2.1 Monte Carlo Moves

PHAISTOS explores the conformational space by applying local Monte Carlo moves to the protein structure. Moves are divided into backbone and side chain moves. All moves work by perturbing one or more internal coordinates. In principle, all internal coordinates are degrees of freedom. However, since bond angles and bond lengths are not treated explicitly by the PROFASI force field, these are constrained by the MC moves to standard values [Engh and Huber, 1991]. Effectively, only dihedral angles are degrees of freedom in the simulations presented here.

This constraint can of course be lifted if the force field include appropriate terms to describe bond angles and bond lengths. For instance this is supported by the OPLS-AA/L force field included in PHAISTOS, which was used in Paper #3.

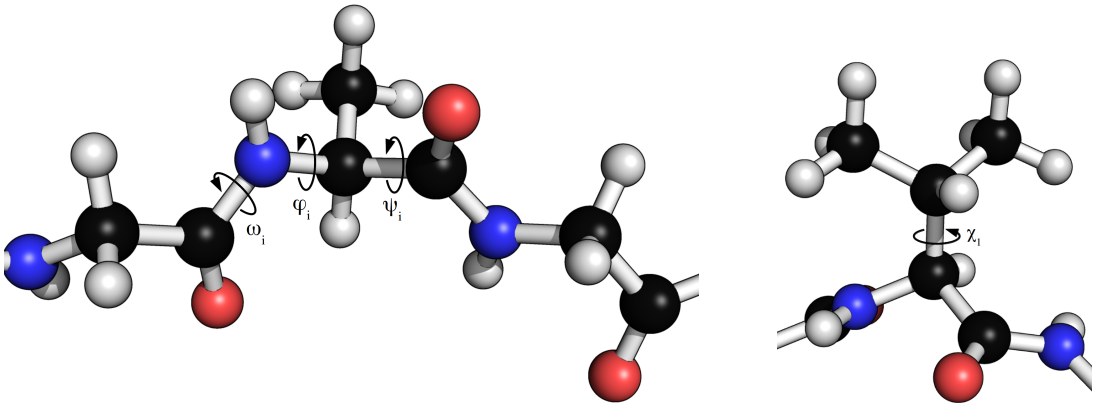
Three different move-types are used in the simulations presented later in this work. These are introduced below. An overview is displayed in Fig. 2.2.

Pivot Move

The pivot move re-samples one dihedral angle of the protein backbone. This usually cause large perturbations since two parts of the protein are rotated relative to each other. As demonstrated later, it is, however, very efficient guiding a folding simulation when biased re-sampling is carried out through TorusDBN or TorusDBN-CS [Boomsma et al., 2008].

CRISP Move

In the CRISP move, a number of consecutive residues are selected (default is 7), and the backbone angles of these are perturbed under the constraint that the end-points are fixed in space [Bottaro et al., 2011]. This move is particularly efficient at exploring dense states, such as native and near-native states. This move also supports biased sampling from TorusDBN and TorusDBN-CS.



(a) Backbone dihedral angles

(b) Side chain dihedral angles

Figure 2.2: The degrees of freedoms in a simulations using the PROFASI force field. The ω, ϕ and ψ dihedral angles on the backbone are shown in (a) for an alanine residue, and the χ_1 dihedral angle for a valine residue is shown in (b).

Side chain Move

Side chain moves can either sample new angles uniformly or biased from via BASILISK [Harder et al., 2010]. Additionally, side chain conformations can be drawn from the Dunbrack-rotamer library [Dunbrack and Cohen, 1997].

Chapter 3

Chemical shifts in a probabilistic framework

This section introduces the formalism for Monte Carlo simulations which includes both physical energy terms as well as a probabilistic energy terms based on experimentally observed chemical shifts. The method presented is not new but has not been published in the form presented here.

Working in a probabilistic framework is a powerful strategy for estimation of unknown parameters, and the intention is to present the equations in the form in which they are implemented in PHAISTOS, so that they can easily be re-implemented in other programs by others. Simulations using the CamShift and ProCS chemical shifts predictors presented later in this thesis employ the equations presented in this chapter.

3.1 Hybrid energy schemes

There are several ways to include experimental observations in simulations, and combine these with known laws of physics. A simplistic approach to this problem is to is to define a hybrid energy by defining a penalty function that describes the agreement between experimental data and data calculated from a proposed model with a physical energy (such as from a molecular mechanics force field). A structure can then be determined, for instance, by minimizing

$$E_{\text{hybrid}} = w_{\text{data}}\ E_{\text{data}} + E_{\text{physical}}. \tag{3.1}$$

where w_{data} is the weight that quantifies the belief in the energy-model E_{data} which defines the agreement between the proposed structure and the experimental data relative to the physical energy.

This concept of using a hybrid energy to determine a protein structure was pioneered by Jack and Levitt who simultaneously minimized a molecular mechanics force field energy and the experimental R-factor for the BPTI protein [Jack and Levitt, 1978]. This approach, however, does not uniquely define neither shape nor weight of E_{data}, and the resulting structure will necessarily depend on these (ill-defined) choices.

Consequently, chemical shifts have been combined with physical energies in a multitude of ways, e.g., weighted RMSD values or harmonic constraints. The groups of Bax and Baker added the chi-square agreement between SPARTA predicted chemical shift values and experimental chemical shifts with an empirical weight of 0.25 to the ROSETTA all-atom energy [Shen et al., 2008]. This methodology was used to determine the structure of 16 small to medium sized proteins.

The CHESHIRE method [Cavalli et al., 2007] uses a hybrid energy function, where a classical energy term is divided by the logarithm of a sum of weighted correlation-coefficients between

SHIFTX calculated chemical shifts and experimental values. Here alpha-hydrogen chemical shifts are weighted by a factor of 18 relative nitrogen and carbon chemical shifts which carry a weight of 1. This hybrid energy is used in the refinement step of the CHESHIRE protocol, and was used to determine the structure of 11 proteins to a backbone RMSD of 1.21 to 1.76 Å relative to the corresponding X-ray or NMR structures.

Vendruscolo and co-workers implemented a "square-well soft harmonic potential", and corresponding molecular gradients and were able to run a chemical shift-biased MD simulation using the CamShift chemical shift predictor [Robustelli et al., 2010]. Subsequently, the trajectory snapshots were re-weighted by multiplying the chemical shift energy term by an empirical weight of 5. Using the empirically optimized balance between energy terms, the native state could be determined from the trajectories for 11 small proteins.

In all cases the parameters and weights of E_{data} had to be carefully tweaked by hand, and it is not clear how to choose optimal parameters. For instance, different types of chemical shifts may (for optimal results) require different weighting, and a brute-force optimization of all parameters is not straight-forward.

3.2 Defining an energy function from Bayes' theorem

The inferential structure determination (ISD) principles introduced by Rieping, Habeck and Nigles [Rieping et al., 2005] defines a Bayesian formulation of Eq. 3.1. The ISD approach rigorously defines the shape and weight of the E_{data} term from the definition of an error model, and allows for the weights to be determined automatically as well. In the following section the equations for an ISD approach are derived for combining the knowledge of experimental chemical shifts with a physical energy.

First remember Bayes' theorem which relates a conditional probability (here A given B) with its inverse:

$$p\left(A|B\right) = \frac{p\left(B|A\right)p\left(A\right)}{p\left(B\right)} \tag{3.2}$$

Now consider a set of chemical shifts $\{\delta_i\}$, the weight for each chemical shift restraint $\{w_i\}$ in the simulation, and finally the structure to be determined, $\mathbf{X}$. This introduces an additional parameter, the weights, which must be determined. These weights describe the belief in the model that relates a structure to a chemical shift. In this case, the most likely structure, $\mathbf{X}$, and optimal choice of $\{w_i\}$ given the set of experimental chemical shifts $\{\delta_i\}$ (via Bayes' theorem) can for instance be found by maximizing:

$$\begin{aligned} p\Big(\mathbf{X}, \{w_i\}\Big|\{\delta_i\}\Big) &= \frac{p\Big(\{\delta_i\}\Big|\mathbf{X}, \{w_i\}\Big)p\Big(\mathbf{X}, \{w_i\}\Big)}{p\Big(\{\delta_i\}\Big)} \\ &\propto p\Big(\{\delta_i\}\Big|\mathbf{X}, \{w_i\}\Big)p\Big(\mathbf{X}, \{w_i\}\Big). \end{aligned} \tag{3.3}$$

Here, the *marginal distribution* of $p\left(\{\delta_i\}\right)$ merely serves as a normalizing factor, and can be neglected. The *likelihood* distribution $p\Big(\{\delta_i\}\Big|\mathbf{X}, \{w_i\}\Big)$ describes the likelihood of the experimental chemical shifts, given a structure, $\mathbf{X}$, and the weights $\{w_i\}$. This requires (1) a forward model to calculate chemical shifts from given structure and (2) an error model that relates the degree of belief in the forward model (that is, the weights) to a probability, based on the difference between experimental and calculated values. Later in this chapter, Gaussian and Cauchy distributions are discussed as error models. The forward model here is a chemical shift predictor, e.g. CamShift, ProCS, etc.

If we assume conditional independence, the *prior* $p\Big(\mathbf{X}, \{w_i\}\Big)$ can be separated as

$$p\Big(\mathbf{X}, \{w_i\}\Big) = p\Big(\mathbf{X}\Big)p\Big(\{w_i\}\Big). \tag{3.4}$$

The two priors, $p\Big(\mathbf{X}\Big)$ and $p\Big(\{w_i\}\Big)$, in brief, describe the distribution of *a priori* meaningful structures (i.e. usually the Boltzmann distribution), and the probability distribution of the weights, respectively. In the following $p(\mathbf{X})$ is simply the Boltzmann distribution, i.e.

$$p(\mathbf{X}) = \frac{1}{Z(T)} \exp\left(-\frac{E(\mathbf{X})}{k_{\mathrm{B}}T}\right) \tag{3.5}$$

where $E(\mathbf{X})$ is the (physical) potential energy of the protein structure, most often calculated using a molecular mechanics force field. k_{B} is the Boltzmann constant and T is the temperature of interest. We need not calculate the partition function, $Z(T)$, because the relative energy landscape is invariant under choice of normalization constant. Note that $p(\mathbf{X})$ also can be introduced via conformational sampling from a biased distribution, such as for example TorusDBN or BASILISK (mimicking the Ramachandran plot and side chain rotamer distributions, respectively). This is discussed later in this chapter.

The prior distribution of the weight parameter $p\Big(\{w_i\}\Big)$ is inherently unknown, except that it is some real number. One such *uninformative prior* could for instance be a flat distribution over the positive real line. This distribution, however, may be biased towards very large numbers. A standard method is to use the Jeffreys' prior, which is a generalization of flat priors, and can be used to model such unknown distributions while introducing only minimal bias. In the one parameter case the Jeffrey's prior is given as

$$p(\theta) \propto \sqrt{\mathbf{I}(\theta)}, \tag{3.6}$$

where $\mathbf{I}(\theta)$ is the *Fisher information* defined (in the one parameter case) as

$$\mathbf{I}(\theta) = \left\langle \left(\frac{\partial}{\partial\theta} \ln p(x|\theta) \right)^2 \right\rangle. \tag{3.7}$$

The corresponding priors for the Gaussian and Cauchy distributions are discussed in the next sections.

3.2.1 Gaussian error model

Selecting an error model is the basic assumption that difference (the error) between a chemical shift calculated from a structure and the corresponding experimentally measured chemical shift, given as $\Delta\delta_i(\mathbf{X}) = \left|\delta_i^{\text{predicted}}(\mathbf{X}) - \delta_i^{\text{experimental}}\right|$, is distributed according to some defined distribution. Following the principle of maximum entropy, the Gaussian distribution is the least biasing distribution, and is the least biasing choice of error model. In this case, the weight parameter introduced in the previous section corresponds to the standard deviation, σ of the Gaussian distribution. For simplicity, it is assumed that the mean of the Gaussian is zero. The total likelihood is then the product of the probability of each $\Delta\delta_i(\mathbf{X})$:

$$\begin{aligned} p\Big(\{\delta_i\}\Big|\mathbf{X}, \{\sigma_i\}\Big) &= \prod_{i=0}^{n} p\left(\Delta\delta_i(\mathbf{X})|\sigma_i\right) \\ &\propto \prod_{i=0}^{n} \frac{1}{\sigma_i} \exp\left(-\frac{\Delta\delta_i(\mathbf{X})^2}{2\sigma_i^2}\right) \end{aligned} \tag{3.8}$$

Next we derive Jeffreys' prior for the uncertainty of a generic Gaussian distribution of the form

$$p(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(x-\mu)}{2\sigma^2}\right). \tag{3.9}$$

Via Eqn. 3.6, this immediately gives us the Jeffreys' prior:

$$\begin{aligned}
p(\sigma) &\propto \sqrt{\left\langle \left(\frac{\partial}{\partial\sigma} \ln p(x|\mu,\sigma)\right)^2 \right\rangle} \\
&= \sqrt{\left\langle \left(\frac{\partial}{\partial\sigma} \ln\left[\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(x-\mu)}{2\sigma^2}\right)\right]\right)^2 \right\rangle} \\
&= \sqrt{\left\langle \left(\frac{(x-\mu)-\sigma^2}{\sigma^3}\right)^2 \right\rangle} \\
&= \sqrt{\int_{-\infty}^{\infty} p(x|\mu,\sigma) \left(\frac{(x-\mu)-\sigma^2}{\sigma^3}\right)^2 dx} \\
&= \sqrt{\frac{2}{\sigma^2}} \propto \frac{1}{\sigma}
\end{aligned} \tag{3.10}$$

Practically, it is impossible to have a separate weight for each individual chemical shift, and the chemical shift of nuclei of the same type thus carry the same weight. The forward model is similar for all nuclei of the same type, so this is somewhat well-justified.

In the following equations, j runs over atom types (e.g. C^α or H^α, etc), and i over residue number. Inserting Eqn. 3.8 and Eqn. 3.10 into Eqn. 3.3, we arrive at a total probability of:

$$\begin{aligned}
p\Big(\mathbf{X}, \{\sigma_j\}\Big|\{\delta_{ij}\}\Big) &\propto p\Big(\{\delta_{ij}\}\Big|\mathbf{X}, \{\sigma_j\}\Big) p\Big(\mathbf{X}\Big) p\Big(\{\sigma_j\}\Big) \\
&\propto \prod_{j=0}^{m}\prod_{i=0}^{n} \frac{1}{\sigma_j} \exp\left(-\frac{\Delta\delta_{ij}(\mathbf{X})^2}{2\sigma_j^2}\right) \exp\left(-\frac{E(\mathbf{X})}{k_\mathrm{B}T}\right) \prod_{j=0}^{m} \frac{1}{\sigma_j} \\
&= \prod_{j=0}^{m} \left(\frac{1}{\sigma_j}\right)^n \exp\left(\sum_{i=0}^{n} -\frac{\Delta\delta_{ij}(\mathbf{X})^2}{2\sigma_j^2}\right) \exp\left(-\frac{E(\mathbf{X})}{k_\mathrm{B}T}\right) \prod_{j=0}^{m} \frac{1}{\sigma_j} \\
&= \prod_{j=0}^{m} \left(\frac{1}{\sigma_j}\right)^{n+1} \exp\left(\sum_{i=0}^{n} -\frac{\Delta\delta_{ij}(\mathbf{X})^2}{2\sigma_j^2}\right) \exp\left(-\frac{E(\mathbf{X})}{k_\mathrm{B}T}\right)
\end{aligned} \tag{3.11}$$

This can be converted to the corresponding hybrid-energy:

$$\begin{aligned}
E_\mathrm{hybrid} &= -k_\mathrm{B}T \ln\left(p\Big(\boldsymbol{X}, \{\sigma_i\}\Big|\{\delta_{ij}\}\Big)\right) \\
&= E(\boldsymbol{X}) + k_\mathrm{B}T \sum_{j=0}^{m} (n+1)\ln(\sigma_j) + k_\mathrm{B}T \sum_{j=0}^{m}\sum_{i=0}^{n} \frac{\Delta\delta_{ij}(\mathbf{X})^2}{2\sigma_j^2}
\end{aligned} \tag{3.12}$$

This expression, except for the term $(n+1)\ln(\sigma)$, is essentially an energy function using harmonic constraints. It is, however, the balance between the two terms which include σ that makes things work. The term $(n+1)\ln(\sigma)$ yields the lowest energy for small values of σ, while the term $\frac{\Delta\delta(\mathbf{X})^2}{2\sigma^2}$ is lower for large values of σ.

Furthermore, the effect of the prior is minute: Using Jeffreys' prior this term is $(n+1)\ln(\sigma)$, whereas using a uniform prior the same term is $n\ln(\sigma)$. Since n is the number of measured chemical shifts of a certain type, the value is usually in the order of ~ 100.

3.2.2 Cauchy error model

Due to numerical instabilities in simulation using the Gaussian error model, a similar model was derived, using a Cauchy distribution as error model. The most notable difference between the Gaussian and Cauchy distributions is that the Cauchy distribution has fatter tails, and thus allows for larger outliers. The differences are discussed in further detail in the Results section in this chapter.

Similarly to Eqn. 3.8, we assume that the location parameter of the Caucy-distribution is zero, and use the scale-parameter, γ as the weight. The total likelihood is then:

$$\begin{aligned} p\Big(\{\delta_i\}\Big|\mathbf{X},\{\gamma_i\}\Big) &= \prod_{i=0}^{n} p\left(\Delta\delta_i(\mathbf{X})|\gamma_i\right) \\ &\propto \prod_{i=0}^{n} \frac{1}{\gamma_i\left[1+\left(\frac{\Delta\delta_i(\mathbf{X})}{\gamma_i}\right)^2\right]} \end{aligned} \tag{3.13}$$

And for the γ parameter of the generic Cauchy distribution of the form

$$p(x|x_0,\gamma) = \frac{1}{\pi\gamma\left[1+\left(\frac{x-x_0}{\gamma}\right)^2\right]}, \tag{3.14}$$

we obtain the following Jeffreys' prior:

$$\begin{aligned} p(\gamma) &\propto \sqrt{\left\langle\left(\frac{\partial}{\partial\gamma}\ln p(x|x_0,\gamma)\right)^2\right\rangle} \\ &= \sqrt{\left\langle\left(\frac{\partial}{\partial\gamma}\ln\left[\frac{1}{\pi\gamma\left[1+\left(\frac{x-x_0}{\gamma}\right)^2\right]}\right]\right)^2\right\rangle} \\ &= \sqrt{\left\langle\left(-\frac{\gamma^2-(x-x_0)^2}{\gamma^3+\gamma(x-x_0)^2}\right)^2\right\rangle} \\ &= \sqrt{\int_{-\infty}^{\infty} p(x|x_0,\gamma)\left(-\frac{\gamma^2-(x-x_0)^2}{\gamma^3+\gamma(x-x_0)^2}\right)^2 dx} \\ &= \sqrt{\frac{1}{2\gamma^2}} \propto \frac{1}{\gamma} \end{aligned} \tag{3.15}$$

Again, it is practically impossible to have a separate weight for each individual chemical shift, and the chemical shift of nuclei of the same type thus carry the same weight. In the following equations, j runs over atom types (e.g. C^α or H^α, etc), and i over residue number. Assembling

the Eqn. 3.13 and Eqn. 3.15 into Eqn. 3.3, we arrive at the total probability of:

$$\begin{aligned} p\Big(\mathbf{X},\{\gamma_j\}\Big|\{\delta_{ij}\}\Big) &\propto p\Big(\{\delta_{ij}\}\Big|\mathbf{X},\{\gamma_j\}\Big)p\Big(\mathbf{X},\{\gamma_j\}\Big) \\ &\propto \prod_{j=0}^{m}\prod_{i=0}^{n}\frac{1}{\gamma_j\left[1+\left(\frac{\Delta\delta_{ij}(\mathbf{X})}{\gamma_j}\right)^2\right]}\exp\left(-\frac{E(\mathbf{X})}{k_{\mathrm{B}}T}\right)\prod_{j=0}^{m}\frac{1}{\gamma_j} \\ &= \prod_{j=0}^{m}\left(\frac{1}{\gamma_j}\right)^{n+1}\prod_{i=0}^{n}\frac{1}{1+\left(\frac{\Delta\delta_{ij}(\mathbf{X})}{\gamma_j}\right)^2}\exp\left(-\frac{E(\mathbf{X})}{k_{\mathrm{B}}T}\right) \end{aligned} \tag{3.16}$$

The associated hybrid energy is then given as:

$$\begin{aligned} E_{\mathrm{hybrid}} &= -k_{\mathrm{B}}T\ln\left(p\Big(\boldsymbol{X},\{\gamma_i\}\Big|\{\delta_{ij}\}\Big)\right) \\ &= E(\mathbf{X})+k_{\mathrm{B}}T\sum_{j=0}^{m}(n+1)\ln(\gamma_j)+k_{\mathrm{B}}T\sum_{j=0}^{m}\sum_{i=0}^{n}\ln\left[1+\left(\frac{\Delta\delta_{ij}(\mathbf{X})}{\gamma_j}\right)^2\right] \end{aligned} \tag{3.17}$$

3.2.3 Marginalization of Weighting parameter

A third option also explored here, is the removal of the weight parameter by projection. This procedure is known as *marginalization*, and is carried out by integrating over all values of the weight parameter. While integration is straight-forward for the Gaussian error-model, the similar expression for the Cauchy distribution does not integrate easily, and the Cauchy-model was not investigated here. From the joint probability distribution in Eqn. 3.11 we obtain the following:

$$\begin{aligned} p_{\mathrm{marginal}}\Big(\mathbf{X}\Big|\{\delta_{ij}\}\Big) &= \int_0^\infty p\Big(\{\delta_{ij}\}\Big|\mathbf{X},\{\sigma_j\}\Big)p\Big(\mathbf{X}\Big)p\Big(\{\sigma_j\}\Big)d\sigma \\ &= \int_0^\infty\prod_{j=0}^{m}\left(\frac{1}{\sigma_j}\right)^{n+1}\exp\left(\sum_{i=0}^{n}-\frac{\Delta\delta_{ij}(\mathbf{X})^2}{2\sigma_j^2}\right)\exp\left(-\frac{E(\mathbf{X})}{k_{\mathrm{B}}T}\right)d\sigma \\ &= \prod_{j=0}^{m}\left(\sum_{i=0}^{n}\Delta\delta_{ij}(\mathbf{X})^2\right)^{n/2}\exp\left(-\frac{E(\mathbf{X})}{k_{\mathrm{B}}T}\right) \end{aligned} \tag{3.18}$$

The hybrid energy associated with the marginalized probability is then given as:

$$\begin{aligned} E_{\mathrm{hybrid}} &= -k_{\mathrm{B}}T\ln\left(p_{\mathrm{marginal}}\Big(\boldsymbol{X}\Big|\{\delta_{ij}\}\Big)\right) \\ &= E(\mathbf{X})+\frac{n}{2}\sum_{j=0}^{m}\ln\sum_{i=0}^{n}\Delta\delta_{ij}(\mathbf{X})^2 \end{aligned} \tag{3.19}$$

3.2.4 Soft Square-Well Energy Function

The last type of hybrid energy term explored here, is a potential designed specifially for molecular dynamics simulations biased by the CamShift predictor [Robustelli et al., 2009, Robustelli et al., 2010]. In this case, the hybrid-energy is given as:

$$E_{\mathrm{hybrid}} = E(\mathbf{X})+\alpha E_{\mathrm{CS}}(\mathbf{X},\{\delta_{ij}\}), \tag{3.20}$$

where $E_{\mathrm{CS}}(\mathbf{X},\{\delta_{ij}\})$ is an empirically derived penalty function that has been demonstrated through simulations to work well for protein structure determination. α is a weight parameter

which was set to 1 during simulation. This penalty function is termed a "soft-square harmonic well", and given by:

$$E_{\mathrm{CS}}(\mathbf{X}, \{\delta_{ij}\}) = \sum_{j=0}^{m} \sum_{i=0}^{n} E_{ij}, \tag{3.21}$$

with

$$E_{ij} = \begin{cases} 0 & \text{if } \Delta\delta_{ij}(\mathbf{X}) < n\epsilon_j \\ \left(\frac{\Delta\delta_{ij}(\mathbf{X}) - n\epsilon_j}{\beta_j}\right)^2 & \text{if } n\epsilon_j < \Delta\delta_{ij}(\mathbf{X}) < x_0 \\ \left(\frac{x_0 - n\epsilon_j}{\beta_j}\right)^2 + \gamma \tanh \frac{2(x_0 - n)(\Delta\delta_{ij}(\mathbf{X}) - x_0)}{\gamma\beta_j^2} & \text{if } x_0 \leq \Delta\delta_{ij}(\mathbf{X}). \end{cases} \tag{3.22}$$

where the parameters, $n\epsilon_j, x_0, \beta_j$ and γ have been empirically adjusted. The potential has a flat bottom, with the width of $n\epsilon_j$. The flat bottom corresponds to the expected standard deviation of CamShift, to avoid overfitting in the simulation. The penalty function grows harmonically until a cut-off of x_0 and follows a somewhat flat hyperbolic tangent function after this. While there is no substantial theoretical backing

3.3 Sampling strategy for weight parameters

Since the nuisance parameters of the energy functions are unknown, they too must be sampled. The move used to update the value of the nuisance parameters must obey detailed balance:

$$p\left(w \to w'\right) = p\left(w' \to w\right) \tag{3.23}$$

The simplest Monte Carlo move is simply adding a number from a normal distribution with $\mu = 0$, this clearly obeys detailed balance, since the distribution is symmetric. For the weight parameters, γ and σ, of the Cauchy and Gaussian distributions, respectively, we found a variance of 0.05 in the normal distributed move to converge quickly and stably.

3.3.1 Molecular mechanics force field

One reasonable prior distribution for protein structure, $p(\mathbf{X})$, is the Boltzmann distribution, e.g.:

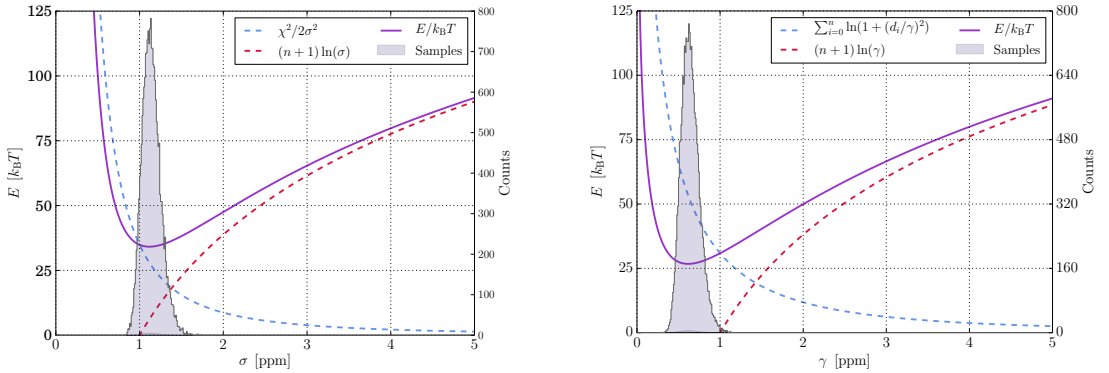
$$p(\mathbf{X}) \propto \exp\left(\frac{-E}{k_{\mathrm{B}}T}\right) \tag{3.24}$$

where E is the energy of the structure, $\mathbf{X}$ and k_{B} and T are Boltzmann's constant and the temperature, respectively. The energy of the structure is in this context usually approximated by a molecular mechanics force field that is taylor-made for protein simulations. PHAISTOS currently supports two different protein force field: The OPLS-AA/L force field with a GB/SA solvent term, and the coarse-grained PROFASI force field. The OPLS-AA/L is an all-atom force field with an additional solvation. The PROFASI force field is a coarse-grained force-field which assumes fixed bond-lengths and angles and furthermore has a very aggressive 4.5 Å cut-off of long-range interaction terms.

3.4 Results

3.4.1 Results – sampling of weight parameters

Figure 3.1 show a histogram of 100,000 sampled values of γ and σ for the NMR structure of Protein G (PDB-id: 2OED). No structural moves were used, and the results are thus temperature independent since the physical energy is constant. A total of 55 C^{α} experimental chemical shifts



(a) Gaussian distribution

(b) Cauchy distribution

Figure 3.1: Sampling of σ and γ for 2OED for Ca-chemical shifts. In this example $n = 55$ and $\chi^2 = 69.7$. Sampled values of the weight parameters clearly cluster around the minimum of the energy function.

were used in this example (RefDB-id: 2575), and CamShift was used to calculate the chemical shifts. The initial values of σ and γ was 10.0, in order to demonstrate the stable convergence using the simple move.

In both simulations, the sampling algorithm converges sampling around the minimum of the energy function. In both cases, these minima are in very good agreement with the values calculated by the test set that was used to validate the performance of CamShift. The largest sampled bins are centered on $\sigma = 1.26$ ppm and $\gamma = 0.63$ ppm for the Gaussian and Cauchy distributions, respectively. These number can be compared to the maximum likelihood estimates (MLE) obtained on the 7 protein benchmark set used to determine the accuracy of Camshift. Here the values are $\sigma = 1.3$ ppm and $\gamma = 0.7$ ppm for the Gaussian and Cauchy distributions, respectively.

3.4.2 Performance of energy functions

Here folding simulation using 11 different variations of the energy function derived and mentioned previously are compared. All energy functions have been implemented in the CamShift module in PHAISTOS, which was also used to run all simulations. The test were carried out on Protein G and the engrailed homeodomain (ENHD). The reference structures were the structures 2OED and 1ENH. An overview of the different simulation types can be found in Table 3.1. For each energy function, 20 independent simulations were carried out for a total of 50,000,000 MC steps each. Each simulation was initialized from a different random, extended strand. Maximum likelihood estimated (MLE) values of the σ and γ weight parameters estimated take from the 7-protein test set reported in reference [Kohlhoff et al., 2009]. For simulations where the weight parameter was sampled, an additional 500,000 Monte Carlo steps were carried out corresponding to the extra moves required to sample this weight (the computational overhead of these 500,000 moves is negligible). Chemical shifts were calculated using the CamShift module. All simulations used the PROFASI force field and sampling from either TorusDBN or TorusDBN-CS.

In two simulations, the bias was removed from the simulation, which corresponds to an unbiased simulation. Two reference simulations were carried out with no chemical shift energy-function, in order to analyze the effect of sampling from TorusDBN and the effect of the PROFASI force field. The simulations used a mix of 40% biased CRISP-moves, 10% biased pivot moves and 50% uniform side chain moves. The simulation was carried out in the multicanonical ensemble via MUNINN. Minimum and maximum β-values were set to 0.3 and 1.05, and the temperature was set to 300K. In all simulations, the number of threads which had samples below thresholds

Table 3.1: Protocols used in the comparison of energy functions and success rates.

Energy type	Weight	TorusDBN-mode	Sampling Bias	Correct sampling[a]	Correct scoring[b]
Gauss	Fixed/MLE	Torus	Biased	20/20	2/6
Gauss	Sampled	Torus	Biased	0/0	0/0
Cauchy	Fixed/MLE	Torus	Biased	20/12	7/4
Cauchy	Sampled	Torus	Biased	20/5	6/1
Cauchy	Sampled	Torus-CS	Biased	20/20	4/2
Cauchy	Sampled	Torus	No bias	0/0	0/0
Cauchy	Sampled	Torus-CS	No bias	0/0	0/0
Square-well	Fixed	Torus	Biased	1/2	1/0
Marginalized	N/A	Torus	Biased	7/17	1/8
No CS	N/A	Torus	Biased	0/0	0/0
No CS	N/A	Torus-CS	Biased	8/10	2/0

[a] Number of threads with a CA-RMSD of < 5 Å (using all residues). Listed as xx for Protein G and yy for ENHD, i.e. xx/yy.
[b] Number of threads where the lowest energy sample has a CA-RMSD of < 3 Å (using all residues). Listed as xx for Protein G and yy for ENHD, i.e. xx/yy.

of 5, 3, 2 and 1 Å CA-RMSD from the crystal structure was recorded. Similarly, the number of threads in which the lowest energy structure was below thresholds of 5, 3, 2 and 1 Å CA-RMSD from the crystal structure was recorded. These figures are used to analyze whether sampling or correct energy scoring is are limiting factors in the particular simulations. The energy was calculated as the PROFASI energy multiplied by $k_{\mathrm{B}}T$ plus the chemical shift energy term plus the log-likelihood calculated from TorusDBN. An overview of these results can be seen in Fig. 3.2 (only simulations that had any samples below 5 Å CA-RMSD from the crystal structure are shown).

For both proteins, using a Gaussian model and sampling the σ uncertainty does not lead to meaningful values for σ. In short, PHAISTOS is able to generate a structure which has no difference between experimental and calculated chemical shifts for a certain atom type. Consequently, the value of σ converges to zero, which effectively freezes the structure in the simulation. The simulations in which the move-bias from TorusDBN and TorusDBN-CS was removed did not sample any structures below

For simulations using Gaussian or Cauchy types of energy function all thread had samples below 5 Å CA-RMSD from the crystal structure for Protein G and between 5-20 for ENHD In the simulation using the square-well potential only 1 thread had samples below 5 Å for Protein G and only 2 for ENHD. For the simulation with marginalized weight parameters, the same figures were 7 and 17, respectively. The reference simulations with no chemical shift in the energy function had no samples below 5 Å for biased sampling from TorusDBN, but 8 and 10 threads below 5 Å for biased sampling from TorusDBN-CS.

Comparing the number of threads for which the lowest energy sample was below 3 Å CA-RMSD from the crystal structure. For both proteins, using fixed weights is somewhat better than using sampled weights with the Cauchy distribution. The result for the square-well potential cannot be interpreted to a statistical significance because only one and two threads were close to the correct fold, but one thread correctly identified the folded state below 3 Å CA-RMSD as the lowest energy for Protein G.

In conclusion, the Gauss and Cauchy error models perform well in sampling and scoring. The fixed MLE weights seem to be work equally well to sampling weights for the cauchy distribution, with no substantial differences. The performance of the energy with marginalized weights

generally performed worse in guiding the sampling, but well in scoring samples for ENHD. The square-well potential did not improve the sampling much. The reason why it has previously been shown to work well, might be that it was combined with a better force-field (AMBER03) to which it was specifically designed. One clear conclusion is that it is useful to not remove the bias from TorusDBN, and keeping the TorusDBN-CS bias seems guide folding significantly more. Even though this formally constitutes is double-counting of effect of knowledge about chemical shifts, this practice seemingly has no adverse effects.

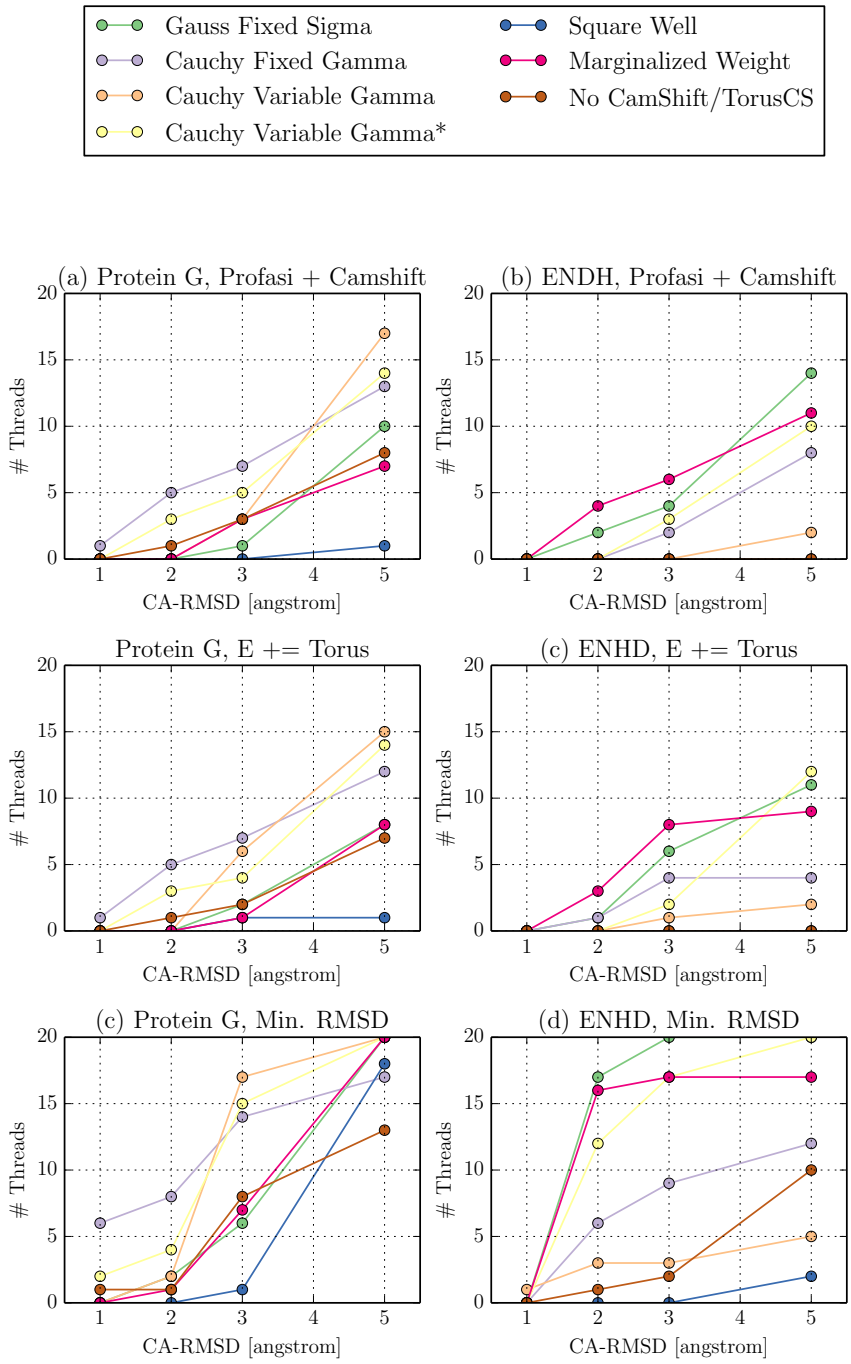


Figure 3.2: Overview of folding simulations using 7 different chemical shift energy types. Sampling was biased by TorusDBN and the PROFASI energy term was used as well. In (a) and (b) the number of threads where the lowest energy samples are under thresholds of 1, 2, 3 and 5 Å CA-RMSD from the crystal structure is plotted. The energy here is calculated as the PROFASI energy multiplied by k_BT plus the chemical shift energy term. In (c) and (d), the log-likelihood from TorusDBN has been added to the total energy. In (e) and (f), the number of threads in which samples are found below under thresholds of 1, 2, 3 and 5 Å CA-RMSD from the crystal structure is plotted. *In this simulation TorusDBN-CS is used instead of TorusDBN.

Chapter 4

Graphical User Interface for PHAISTOS

Setting up simulations in PHAISTOS requires expert knowledge about the program. Firstly, while all modules and settings have reasonable default settings, there are still many things that cannot be specified via default alone, and secondly, the complete list of settings in PHAISTOS is around 2500 options that must be set or taken as default values.

In order to make PHAISTOS more attractive to new users, I wrote a GUI can set up most simulations for most of the simulations covered by this thesis. The GUI for PHAISTOS is aptly named Guistos and is written in Python 2.x using TkInter.

Using the GUI the user is only presented with the three most basic choices for setting up the simulation. These are (1) choice of energy terms, (2) type of Monte Carlo simulation and finally (3) a selection of Monte Carlo moves. A screenshot of Guistos can be seen in Fig. 4.1. Setting up these via Guistos is discussed below.

Energy Options

Firstly, the Energy Options section allows the user to select the molecular mechanics force field. Currently two force fields are supported in PHAISTOS, which are the OPLS-AA/L force field with a GB/SA solvent model, and the PROFASI coarse grained force field. Use of the PROFASI force field requires the Monte Carlo moves to restraint the bond angle and lengths in the protein to Engh-Huber standard values. This is automatically done if the PROFASI force field is selected. Conversely, the OPLS-AA/L force field includes energy terms for bond angles and lengths and these are degrees of freedom in the simulation if the OPLS-AA/L force field is selected.

Additionally, the Energy Options section allows the user to add restrains from one type spectroscopic data. Currently energy terms based on CamShift 1.35 and ProCS are supported. These options requires a NMR-STAR formatted file containing experimental chemical shifts.

Monte Carlo Options

This section allows the user to select the four types of Monte Carlo simulation offered by PHAISTOS and the only the most basic options to set up that particular simulation: Metropolis-Hastings offers the choice of a constant temperature (in Kelvin). Muninn and Simulated Annealing offer the choice of a temperature range (in Kelvin), and additionally Muninn offers the choice between multicanonical or $1/k$ sampling. Greedy Optimization does not offer any customizable option.

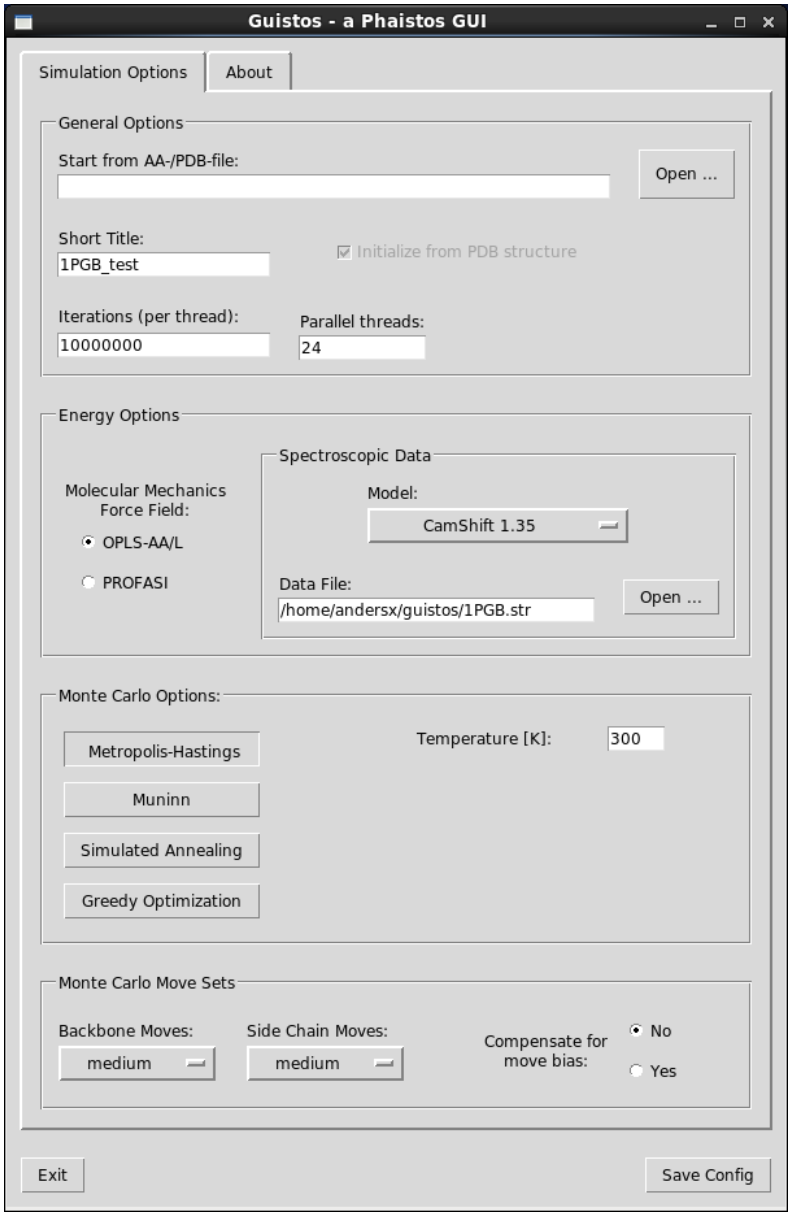


Figure 4.1: Screenshot of Guistos

Monte Carlo Move Sets

Selecting a good mix of the different Monte Carlo moves offered by PHAISTOS can significantly speed up convergence of a simulation, compared to using an inferior move set. Choosing a good set of moves is in the opinion of this author currently somewhere in between black art and sheer luck, and requires a good deal of experience with simulations in PHAISTOS.

To make it easier for new users, three move sets have been predefined using the experience of this author. These are named "small", "medium" and "large". The "small" move set is intended for uses such as refinement or sampling around a compact native state, while the "medium" move set is intended for folding simulations that start from extended, but are expected to also sample a native state, and finally the "large" move set is intended for sampling conformational space quickly, but will have problems with sampling compact structures. All move sets sample from TorusDBN (backbone angles) and BASILISK (side chain angles), and an option to remove this bias is also present.

Using Guistos

Guistos is freely released under the open source two-clause BSD-license, and can be downloaded from `https://github.com/andersx/guistos/`. After specifying all relevant settings in the Guistos window, a configuration-file is saved by pressing the "Save Config" button. A simulation in PHAISTOS can the be executed via the following command:

```
1 ./phaistos --config-file my_simulation.config
```

Chapter 5

Prediction of Protein Chemical Shifts

While the relationship between NOE restraints and the underlying protein structure is clear, the relationship between chemical shifts and the structure is less clear. Several programs, however, exists which are able to predict protein chemical shifts given a protein structure. Typically, these chemical shift predictors are parametrized from empirical fits between experimental crystal structures of proteins to their corresponding measured NMR chemical shifts. Popular programs that employ such empirical include SHIFTX, SPARTA+, SHIFTS and CamShift [Neal et al., 2003, Shen and Bax, 2010, Ösapay and Case, 1991, Kohlhoff et al., 2009]. These programs use functional forms that decompose the chemical shift into additive, independent terms. The accuracy of these fits are inherently limited by the availability and accuracy of empirical data. A similar program, CheShift, exists, in which the functional forms are interpolated from a large database of QM calculation on representative peptide conformations [Vila et al., 2009]. The authors, however, have not been willing to share the code, but exists as a web-service which allow aplha-carbon and beta-carbon chemical shift calculations.

We have recently explored using quantum mechanics to derive chemical shifts from protein structures. Our amide-proton chemical shift predictor is discussed in our paper #3 in Appendix A. Briefly, in the amide proton-only version of ProCS [Christensen et al., 2013], the chemical shift is calculated as a sum of several independent terms [Parker et al., 2006]:

$$\delta_{\mathrm{H}} = \delta_{\mathrm{BB}}(\phi, \psi) + \Delta\delta_{\mathrm{HB}} + \Delta\delta_{\mathrm{rc}} \tag{5.1}$$

where $\delta_{\mathrm{BB}}(\phi, \psi)$ chemical shift dependence on the backbone angles, $\Delta\delta_{\mathrm{HB}}$ is a sum over 3 different contributions due to hydrogen bonding and $\Delta\delta_{\mathrm{rc}}$ is the perturbation due to magnetic field from aromatic side chains [Christensen et al., 2011]. All terms are parametrized by QM methods by fitting the terms to QM calculations on model systems. The ring current contribution term is discussed in detail in publication #1 in the appendix.

As we show in the publication, structures generated using amide-proton chemical shift restraints from ProCS have hydrogen bonding geometries that are in substantially better agreement with experimental X-ray structures and back-calculated experimentally measured spin-spin coupling constants, compared to using CamShift as predictor or no chemical shifts in the simulation. The accuracy of the amide proton-only version of ProCS is lower than SHIFTX, SPARTA+, SHIFTS or CamShift, when experimental protein structures are used as input, but we show that this is likely due to inaccuracies in the experimental coordinates.

Similar to the approximation above, we have made a predictor for all backbone and beta-carbon. In the backbone atom version of ProCS, the chemical shift is calculated as:

$$\delta = \delta_{\mathrm{BB}} + \Delta\delta_{\mathrm{HB}} + \Delta\delta_{\mathrm{rc}} \tag{5.2}$$

where δ_{BB} is due to dihedral bond angles in the residue and the neighboring residues, and $\Delta\delta_{\mathrm{HB}}$ and $\Delta\delta_{\mathrm{rc}}$ are implemented similarly to those of the amide proton-only version of ProCS. To accurately calculate the dependence of angles and neighboring residues on carbon and nitrogen chemical shift, we found an accurate description to be:

$$\delta_{\mathrm{BB}} = \delta_i(\phi_i, \psi_i, \{\chi_i\}) + \Delta\delta_{i-1}(\phi_{i-1}, \psi_{i-1}, \{\chi_{i-1}\}) + \Delta\delta_{i+1}(\phi_{i+1}, \psi_{i+1}, \{\chi_{i+1}\}), \tag{5.3}$$

where δ_i takes into account, the chemical shift due to the ϕ_i, ψ_i and $\{\chi_i\}$ angles on the i'th residue, and $\Delta\delta_{i-1}$ and $\Delta\delta_{i+1}$ takes into account the perturbation due to the neighboring residue conformation and residue type.

The three terms in δ_{BB} are interpolated through exhaustive scans over all possible conformations of tri-peptides. To set up the massive number of QM calculations, the FragBuilder Python API was created (see Paper #4). FragBuilder is an Python API that makes it possible to easily generate peptide conformations, either via manual definition of dihedral angles or sampling via the BASILISK library [Harder et al., 2010]. Using the OpenBabel Python API [O'Boyle et al., 2011], it is furthermore possible to perform molecular mechanics optimizations and write coordinate files in nearly 100 different formats. FragBuilder provides convenient wrappers and classes for such operations, and only few lines of code are generally needed for generating an input-file.

The FragBuilder Python API was used to generate the more than 2,000,000 peptide structures used to generate the database. The peptide structures were optimized using the PM6 semi-empirical QM method, and QM chemical shifts were calculated at the OPBE/6-31G(d,p) level using a polarizable continuum model to model an embedding environment. The resulting tables of chemical shifts were collected and stored in files in Numpy's binary .npz-format [Oliphant, 2006].

The predictor is programmed into a separate module for PHAISTOS (in C++). The program loads the Numpy-arrays into the memory and uses existing code to read coordinates and angles. These tables are roughly 10GB for each nucleus, so the current version of ProCS requires about 64GB of RAM for predicting backbone atom and beta-carbon chemical shifts efficiently.

5.1 Initial Results

The code is currently not ready for use in simulations, other than for testing purposes, due to the massive memory requirements, and parallelization is not yet complete, so no results in this respect can be presented here.

Initial test show that calculating chemical shifts via the ProCS module is about 5 times faster than the CamShift energy term in PHAISTOS and roughly same speed as the PROFASI energy term. Note, that the CamShift and PROFASI energy terms use a caching algorithm which effectively means that only terms that depend on atoms that are move during a Monte Carlo move have to be re-calculated each move. An initial cached version of ProCS is around 5 times faster than the non-cached version, and thus faster than the coarse-grained PROFASI force field. Fast evaluation of chemical shifts is crucial for including the chemical shift predictor in the energy function when simulating folding of larger proteins ($>$ 100 amino acids), where the CamShift predictor is currently too slow for our purpose.

We have assessed the accuracy of ProCS for alpha-carbon and beta-carbon atoms by comparison to benchmark QM calculations on an entire proteins. The experimental structures of Protein G and Ubiquitin (PDB-codes: 2OED and 1UBQ, respectively) were protonated using the PDB2PQR webinterface [Dolinsky et al., 2004, Dolinsky et al., 2007]. Additional structures were generated by minimizing the X-ray structures in Tinker with the AMBER, CHARMM22/CMAP and AMOEBA force fields with a GB/SA solvent model. The chemical shifts of the resulting structures were calculated in GAUSSIAN 09 [Frisch et al., 2009] at the OPBE/6-31G(d,p) level with a polarizable continuum solvent model. The results are summarized in table 5.1.

The QM calculations on Ubiquitin are in slightly better agreement with the ProCS predicted number, than the CheShift and CamShift predicted values, based on RMSD and r^2 values. For

Protein G CheShift are and CamShift RMSD values are slightly lower for alpha-carbon, while ProCS has a lower RMSD for beta-carbon. The general trend is that the predictors are comparable in accuracy.

Table 5.1: Comparison of agreement between QM calculation of alpha-carbon and beta-carbon chemical shifts and predicted chemical shifts, for X-ray structures of Ubiquitin and Protein G, and structures minimized with the AMBER, CHARMM22/CMAP and AMOEBA force fields.

	ProCS		CheShift		CamShift	
CA/Ubiquitin	r^2	RMSD	r^2	RMSD	r^2	RMSD
1UBQ (X-ray)	0.754	2.54	0.697	3.63	0.666	2.97
AMBER	0.815	1.93	0.789	3.19	0.763	2.41
CHARMM22/CMAP	0.897	2.78	0.775	2.12	0.827	2.68
AMOEBA	N/A	N/A	0.851	3.94	0.886	2.26
CA/Protein G	r^2	RMSD	r^2	RMSD	r^2	RMSD
2OED (X-ray)	0.894	2.37	0.883	1.66	0.887	2.21
AMBER	0.824	3.02	0.883	1.87	0.883	1.87
CHARMM22/CMAP	0.907	2.60	0.814	2.13	0.839	2.82
AMOEBA	0.914	1.90	0.866	3.84	0.755	2.82
CB/Ubiquitin	r^2	RMSD	r^2	RMSD	r^2	RMSD
1UBQ (X-ray)	0.947	3.44	0.945	3.90	0.941	3.58
AMBER	0.983	1.91	0.965	2.85	0.964	2.54
CHARMM22/CMAP	0.980	2.76	0.971	5.22	0.970	3.34
AMOEBA	N/A	N/A	0.957	6.34	0.950	4.30
CB/Protein G	r^2	RMSD	r^2	RMSD	r^2	RMSD
2OED (X-ray)	0.992	2.87	0.983	2.2	0.983	3.10
AMBER	0.974	2.91	0.982	2.63	0.982	2.63
CHARMM22/CMAP	0.991	2.68	0.979	4.95	0.985	3.08
AMOEBA	0.984	3.83	0.977	6.29	0.977	4.06

Chapter 6

Determined protein structures

This section describes all test-targets which I have attempted to fold using the methodologies presented in the previous chapters. All protein structures, chemical shift and NOE data used in this thesis is available from `https://github.com/andersx/cs-proteins/`.

6.1 Barley Chymotrypsin Inhibitor II

An especially interesting target in this study is the barley chymotrypsin inhibitor II (CI-2). CI-2 is a 63 residue protein which consists of an α-helix which connects via a very flexible handle to a small β-sheet region.

The chemical shifts data supplied by Kaare Theilum (personal communication) was obtained using a fully automated procedure. The ADAPT-NMR [Bahrami et al., 2012] protocol was used to record all necessary NMR data and automatically assign the chemical shifts. Data collection and assignment was completed in only 11 hours with minimal human intervention. As we demonstrate, a structure could be determined computationally from these chemical shifts in only two days running on 12 cores.

6.1.1 Computational methodology

Several folding protocols were tried for this protein. All runs were performed as 72 independent trajectories which ran for 50 mio MC steps (iterations). Sampling was carried out using either TorusDBN or TorusDBN-CS to bias the backbone moves and the PROFASI force field was used in all simulations. One simulations used an experimental version of TorusDBN-CS, supplied by Lars Bratholm, which was trained on only high-resolution X-ray structures (available from `https://github.com/andersx/cs-proteins/`). Three simulations used an energy function based on CamShift using a cauchy distribution with variable γ-weight as energy function. Additionally, three simulations used a potential on the radius of gyration to restrict the sampling to only compact structures [Borg et al., 2009]. Sampling was performed in the multicanonical ensemble with a thermodynamic beta-range from 0.6 to 1.1, corresponding to a temperature range of 272K to 500K. The MC move set was comprised of 40% CRISP moves, 10% pivot moves and 50% uniform side chain moves.

6.1.2 Folding results

Three of the 7 attempted simulation types sample structures close to the experimental X-ray structure 1YPA (here loosely defined as a CA-RMSD < 5 Å for all CA atoms. Results are summarized in table 6.1. Only simulations using chemical shift biased sampling through TorusDBN-CS are able to sample the correct fold.

Table 6.1: Protocols used in the folding of the CI-2 protein and success rates.

Sampling	Force Field	CS Energy	Correct fold[a]	Iterations/day[b]
TORUS-CS + RG[c]	PROFASI	CamShift	13	10×10^6
TORUS-CS	PROFASI	CamShift	15	11×10^6
TORUS	PROFASI	CamShift	0	11×10^6
TORUS-CS + PP[c]	PROFASI	None	4[d]	49×10^6
TORUS-CS[e] + RG[c]	PROFASI	None	0	49×10^6
TORUS-CS	PROFASI	None	0	49×10^6
TORUS	PROFASI	None	0	49×10^6

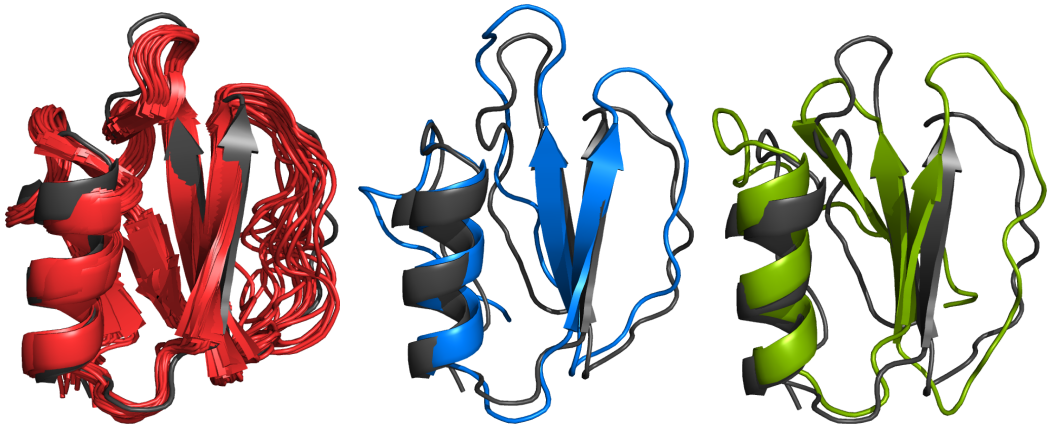
[a] Number of threads with a CA-RMSD of < 5 Å (using all residues).
[b] Numbers are *per* thread.
[c] RG denote the use of an additional radius of gyration potential.
[d] Structures with the lowest energy did not correspond to the native structure in this run.
[e] This run was carried out using TorusDBN-CS trained using only high-quality X-ray structures.

Furthermore, it was noted, that simulations that sample from either TorusDBN or TorusDBN-CS with only the PROFASI force field as energy function do not generate compact structures. To overcome this deficiency, additional simulations were carried out using a radius of gyration potential. In the case of sampling from TorusDBN-CS, the radius of gyration potential is enough to get a few samples with the correct fold. Here four of 72 threads would generate the correct fold, but unfortunately the lowest energy structures were found around 8-11 Å CA-RMSD. Evidently, the PROFASI force field alone is not accurate enough to describe the native CI-2 structure. Three simulations were performed with an energy term based on CamShift in addition the PROFASI force field. Demonstrably, the increased accuracy from a better energy function cause increased sampling around the native state.

Due to a very flexible region of CI-2 (residues 33 to 42), and somewhat flexible tails the residue range used to calculate CA-RMSD values is restricted to residue 4-34,43-63 in the following. All runs were carried out on 3 24-core AMD Opteron 6172 servers running at 2.1 GHz.

A run similar to the most successful was also run carried out on a faster a 12-core Intel X5675 node running at 3.07 GHz (using new random seeds). PHAISTOS input to reproduce these folding simulation is given below.

```
./phaistos --aa-file ci2.aa \
  --iterations 50000000 \
  --threads 12 \
  --monte-carlo-muninn 1 \
  --monte-carlo-muninn-min-beta 0.6 \
  --monte-carlo-muninn-max-beta 1.1 \
  --monte-carlo-muninn-independent-threads 1 \
  --monte-carlo-muninn-weight-scheme multicanonical \
  --backbone-dbn-torus-cs 1 \
  --backbone-dbn-torus-cs-initial-nmr-star-filename ci2.str \
  --energy-profasi-cached 1 \
  --energy-camshift-cached 1 \
  --energy-camshift-cached-star-filename ci2.str \
  --energy-camshift-cached-energy-type 11 \
  --move-backbone-dbn 1 \
  --move-backbone-dbn-weight 0.1 \
```



(a) NMR structure (red) (b) Lowest RMSD structure (blue) (c) Lowest energy structure (green)

Figure 6.1: Structures compared to the X-ray structure 1YPA. All structures are aligned using the residues 12-32,43-52. (a) shows the 3CI2 structure NNR structure. Note the flexible domain which is excluded from the fit-range. (b) Shows the lowest RMSD structure (1.113 ÅRMSD). (c) shows the lowest energy sample (2.76 ÅRMSD).

```
17   --move-backbone-dbn-implicit-energy 1 \
18   --move-crisp-dbn-eh 1 \
19   --move-crisp-dbn-eh-weight 0.4 \
20   --move-sidechain-uniform 1 \
21   --move-sidechain-uniform-weight 0.5
```

This simulation took two days, with a total of 2 out of 12 threads successfully identifying the native structure as having the lowest energy. This simulation yielded a lowest energy structure a 2.76 Å CA-RMSD from the X-ray structure, and a lowest RMSD structure at 1.11 Å. Later, this lowest energy sample was further refined by Lars Bratholm to a CA-RMSD of only 1.1 Å using an additional multibody-multinomial potential of mean force in the energy function [Johansson and Hamelryck, 2013]. This refinement simulation took 24 hours on 8 cores. This structure is displayed in Fig. 6.2.

In conclusion, the data for CI-2 was recorded in merely 11 hours via a fully automated process. A structure comparable to conventional NMR structures could then be determined after 36 hours. After an additional 24 hours, a structure that rivals X-ray structures was further determined by Lars Bratholm.

Figure 6.2: The CI-2 structure refined to 1.1 Å by Lars Bratholm. The refinement was carried out by including a multibody-multinomial poteintial of mean force in the simulation.

Table 6.2: The five small proteins folded using the setup presented in this section, and their RMSD for the lowest energy sample.

Name	Lengh	Type	PDB	RefDB	RMSD-range	Final RMSD
Protein G	56	a/b	2OED	2575	All	1.0
Engrailed Homeodomain	61	B	1ENH	15536	8-53	1.1
FF Domain	71	a/b	1UZC	5537	11-67	10.2
Ubiquitin	76	a/b	1UBI	17769	1-70	3.8
CI-2	63	a/b	1YPA	N/A[a]	4-34,43-63	2.6[b]

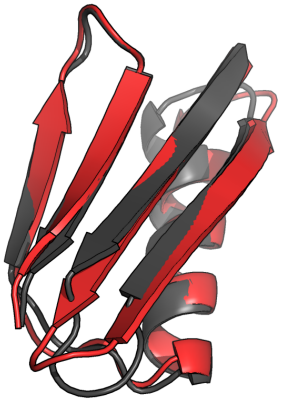
[a] Using automatically assigned data obtained from Kaare Theilum (personal communication – see `https://github.com/andersx/cs-proteins/`). [b] The number reported is discussed in section 6.1.2.

6.2 Folding of small proteins (<100 AA)

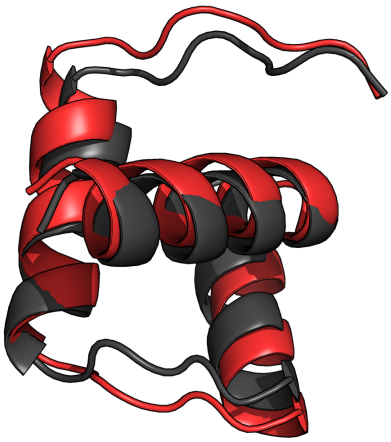
A test set of 5 small proteins were folded using the code. The results are summarized in table 6.2. The test set is a diverse set of structures with different contents of alpha-helix and beta-sheet conformations. The settings are similar to the ones used to fold the CI-2 structure mentioned in the previous section, except that the Protein G, Ubiquitin, FF Domain and Engrailed Homeodomain (ENHD) simulations used a chemical shift energy based on a Gaussian distribution with fixed weights (–energy-camshift-cached-energy-type 3), and not based on a Cauchy distribution (–energy-camshift-cached-energy-type 11). Total energy was calculated as the PROFASI force field energy plus the CamShift energy term based on a Gaussian distribution with fixed weights plus the likelihood from TorusDBN-CS. Protein G and ENHD structures could be determined very reliably to CA-RMSDs of 1.0 Å and 1.1 Å from the experimental structures, respectively. The lowest energy structures are presented in Fig. 6.3a and 6.3b.

For the FF Domain, a folded state with a lower energy than the native state was located. A state corresponding to the correct fold was consistently being sampled in most threads, but the lowest energy stat was a misfold, where an alpha-helix towards the C'-end is packed wrongly. This result suggests, that the combination of the PROFASI force field and the chemical shift energy from CamShift and TorusDBN-CS does not always discriminate the potential energy surface with sufficient accuray. The energy from CamShift (and thus the chemical shift RMSD values, since the energy function was a Gaussian distribution) was comparable between samples around the correct fold and the lowest energy mis fold. The lowest RMSD structre (3.2 Å) had a CamShift energy of 803 kcal/mol, while the lowest energy structure had a CamShift energy of 797 kcal/mol. The lowest energy misfold is displayed in Fig. 6.3c. In the Ubiquitin simulations, the lowest energy conformations were not in exceptional agreement with the experimental structure with a CA-RMSD of 3.8 Å- see Fig. 6.3d. Again, this must be attributed to lack of "funneling" of the energy landscape around the native state, since sampling evidently is performed close to this state.

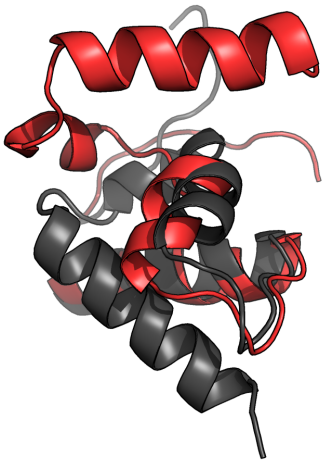
Collectively, these result show, that sampling from TorusDBN-CS in PHAISTOS is indeed very efficient, but better energy functions are required in some cases. In one case, the CamShift energy term had a lower energy by 6 kcal/mol for a misfold, than for a sample close to the native state. Another option would be using a better molecular mechanics force-field. PHAISTOS already supports the OPLS-AA/L but using this would increase simulation times by more than one order of magnitude, and would be unacceptable for simulations on larger structures.



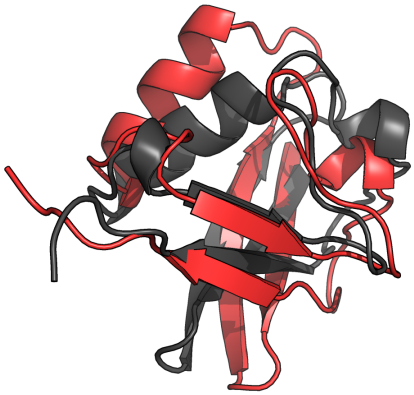
(a) Protein G, 1.0 Å.



(b) Engrailed Homeodomain, 1.1 Å.



(c) FF Domain, misfold.



(d) Ubiquitin, 3.8 Å

Figure 6.3: The lowest energy structures found for four different proteins (red). Superimposed on corresponding X-ray structures (grey). The FF Domain structures in (c) is aligned using only residues 1-40 to emphasize the misfold.

6.3 Folding of larger proteins (>100 AA)

This section presents folding results on a set of larger proteins (>100 AA) with known structures. It is worth to note, that using sparse NMR data, only three structures >200 residues have been determined: Alg13 (201 AA), Rhodopsin (225 AA) and MBP (376 AA) using the ROSETTA program with the "resolution-adapted structural recombination" (RASREC) protocol [Lange and Baker, 2012, Lange et al., 2012].

Alg13 was solved using backbone chemical shifts, and only 52 NOE restraints, to an CA-RMSD of 4 Å to the experimental NMR structure. Rhodopsin was folded to an CA-RMSD of 1.9 Å to the X-ray structure using 215 NOE restraints, backbone chemical shifts chemical shifts and RDCs. The MBP protein is a two-domain protein of 376 residues. MBP was folded to an RMSD of 3.6 Å using 1235 NOE restraints, backbone chemical shifts chemical shifts and RDCs. The NOEs corresponded to 55% yield of restraints, which, for the most part, were not automatically assigned. An attempt to use only automatically assigned NOEs yielded 455 restraints, which corresponds to a yield of 20%. Using these, however, the MBP structure could only be determined to a total CA-RMSD of 12.3 Å. The N-terminal domain was converged to 2.7 Å, but the C-terminal domain and the angle between the two domains was incorrectly folded.

Langer *et al.* have demonstrated that by using a special side-chain labeling scheme a few NOE restraints (around 150-250) can be automatically assigned, and these are generally enough to fold the structures using a ROSETTA protocol[Lange et al., 2012]. The scheme is a "ILV-labeling" scheme, where the methyl groups of isoleucine, leucine and valine side-chains are selectively labeled with ^{13}C and ^{1}H isotopes. These groups are commonly found in the core region of the protein and these methyl groups will generally be in contact with each other, thus being able to provide valuable NOE distance restraints. This corresponds to only assigning 10-20% of the full spectrum.

From the structures in the study by Langer *et al.*, only five structures consist of one chain only, and only those could be simulated in PHAISTOS. These five structures were selected into the test-set used here, and additionally Prolactin and the Top7 proteins were added. The ILV-data used by Langer *et al.* could only be obtained through correspondence with the authors for Rhodopsin. For all other proteins, synthetic NOE contacts were generate by simulating a synthetic spectrum. An overview of the proteins and the number of synthetic NOE restraints can be found in Table 6.3.

Table 6.3: Folded structure.

Name	Lengh	Type	PDB	BMRB	RMSD-range	#NOEs	RMSD [Å]
Top7	120	a/b	2MBL	19404	5-104	62	2.1
MSRB	151	a/b	3E0O	17008	36-105	170	N/A
WR73	183	a/b	2LOY	16833	1-36,66-181	215	N/A
HR4660B	174	a/b	2LMD	1870	16-162	68	N/A
Rhodopsin	219	B	2KSY	16678	All	195	2.5
Prolactin	199	B	1RWS	5599	6-183	68	3.5
Savinase	269	a/b	1WVN	Note[a,b]	Note[b]	270	2.9
MBP	376	a/b	1EZ9	6807	All	1054	N/A

[a] Evolutionary distance constraints from the EVFold were used in this case.
[b] Available from: `http://github.com/andersx/cs-proteins/`

6.3.1 Folding protocol

The folding simulation settings were similar to those used to fold small proteins, with the exception that the CamShift energy term was too slow to be used in practice. The additional NOE distance restraint term used a flat-bottom potential with a width of 4 Å around the equilibrium distance, and a quadratic potential outside this range. This was done using the existing NMR inference module in PHAISTOS.

However, using this potential turned out to be quite problematic. Once a distance restraint was fulfilled, the simulation would in most cases never break the contact again. Consequently, an empirical factor of 1/128 was multiplied onto the NOE energy. This factor was determined by running simulations on the Top7 structure with weights from $1/2^1$ to $1/2^{10}$ Unfortunatly, due to this problem, no good structures for MSRB, WR73, HR4660B and MBP could be located. After a few 1,000,000 steps the structures located local minima which fulfilled a number of distance restraints, but it was impossible to escape these minima. The Top7 structure folded to an RMSD of 2.1 Å. This result, however, is not surprising, since Top7 has been shown to fold using only the PROFASI force field. The Prolactin and Rhodopsin structures converged to structures at 8.5 and 7.8 Å RMSD from the X-ray structures.

The settings to run the simulations are displayed below:

```
./phaistos --aa-file rhodopsin.aa \
  --iterations 50000000 \
  --threads 72 \
  --monte-carlo-muninn 1 \
  --monte-carlo-muninn-min-beta 0.6 \
  --monte-carlo-muninn-max-beta 1.1 \
  --monte-carlo-muninn-independent-threads 1 \
  --monte-carlo-muninn-weight-scheme multicanonical \
  --backbone-dbn-torus-cs 1 \
  --backbone-dbn-torus-cs-initial-nmr-star-filename \
                                      rhodopsin.str \
  --energy-profasi-cached 1 \
  --energy-isd-dist 1 \
  --energy-isd-dist-likelihood square_well \
  --energy-isd-dist-data-filename noe_ilv.txt \
  --energy-isd-dist-sample-gamme 0 \
  --energy-isd-dist-sample-sigma 0 \
  --energy-isd-dist-weight 0.0078125 \
  --move-backbone-dbn 1 \
  --move-backbone-dbn-weight 0.08 \
  --move-backbone-dbn-implicit-energy 1 \
  --move-crisp-dbn-eh 1 \
  --move-crisp-dbn-eh-weight 0.42 \
  --move-sidechain-uniform 1 \
  --move-sidechain-uniform-weight 0.5
```

6.3.2 Refinement protocol

Due to the low efficiency of the NOE code for large structures, a new NOE module was written for PHAISTOS. In this module, the potential from the ROSETTA RASREC protocol was used [Lange et al., 2012]. In brief, this is also a flat-bottom potential, but with a linear penalty, rather than quadratic, outside the flat area. This was done in order to allow more contacts to be broken throughout the simulation in order to enhance conformational sampling. Additionally,

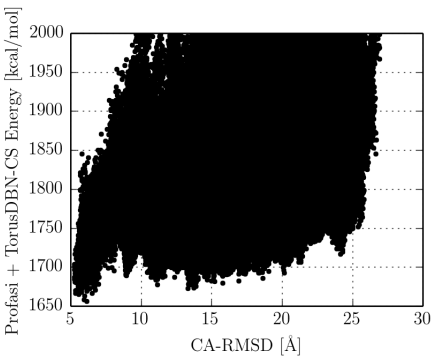
the module only has a certain fraction of all restraints active at a time. A Monte Carlo move was created which turned off one random, active NOE restraint and activated one random, deactivated restraint. The resulting energy difference was subtracted as a move-bias, in order to force a 100% acceptance rate for this move. This was done, because the energy difference between and active restraint (which is usually close to zero) and an inactive restraint (usually a large number) caused this move to have a low acceptance rate.

Using the new NOE module, a refinement on the lowest energy structures in the Prolactin and Rhodopsin simulations were carried out.

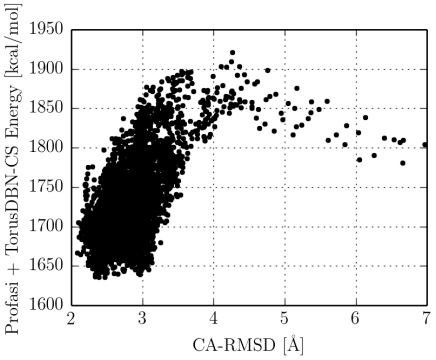
The new module proved very efficient in further minimizing the energy. Fig. 6.4 shows the resulting structures and energy/RMSD landscapes from the refinements and folding simulations on Rhodopsin. The final RMSD after refinement was 2.5 Å for Rhodopsin, compared to 7.8 Å before refinement. For Prolactin, the same numbers were 3.5 Å and 8.5 Å, respectively. The reason for the higher RMSD for Prolactin, compared to Rhodopsin is a flexible handle with no NOE restraints. The structure of this handle is thus determined by the PROFASI force field and TorusDBN-CS, which apparently does not agree well with the experimental structure in this case - this can be seen from Fig. 6.5.

The command line to run the refinement is given below:

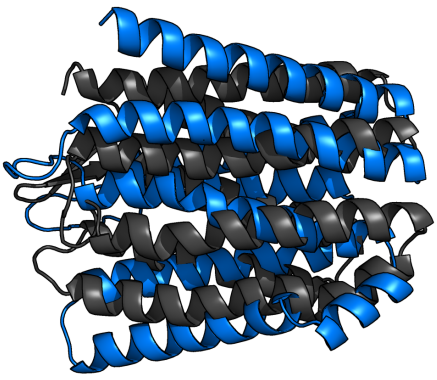
```
./phaistos --pdb-file rhodopsin_lowest_energy1.pdb \
  --init-from-pdb 1 \
  --iterations 5000000 \
  --threads 4 \
  --monte-carlo-muninn 1 \
  --monte-carlo-muninn-min-beta 0.6 \
  --monte-carlo-muninn-max-beta 1.1 \
  --monte-carlo-muninn-independent-threads 1 \
  --monte-carlo-muninn-weight-scheme multicanonical \
  --monte-carlo-muninn-weight-scheme-use-energy2 1 \
  --backbone-dbn-torus-cs 1 \
  --backbone-dbn-torus-cs-initial-nmr-star-filename \
                                      rhodopsin.str \
  --energy-profasi-cached 1 \
  --energy2-noe 1
  --energy2-noe-active-restraints 140
  --energy2-noe-seamless 1
  --energy2-noe-contact-map-filename noe_ilv.txt
  --move-none 1\
  --move-none-weight 0.005 \
  --move-crisp-dbn-eh 1 \
  --move-crisp-dbn-eh-weight 0.5 \
  --move-semilocal-dbn-eh 1 \
  --move-semilocal-dbn-eh-weight 0.25 \
  --move-sidechain-rotamer 1 \
  --move-sidechain-rotamer-weight 0.25
```



(a) Energy-scoring during folding stage.



(b) Energy-scoring during refinement stage.



(c) Folding stage lowest energy sample (blue). 7.8 Å CA-RMSD.

(d) Refinement stage lowest energy sample (blue). 2.5 Å CA-RMSD.

Figure 6.4: (a) displays the energy scoring during the folding stage of Rhodopsin, and (b) the same statistics during the refinement stage. (c) displays the lowest energy structure after the folding stage, and (d) the lowest energy structure after the refinement stage.

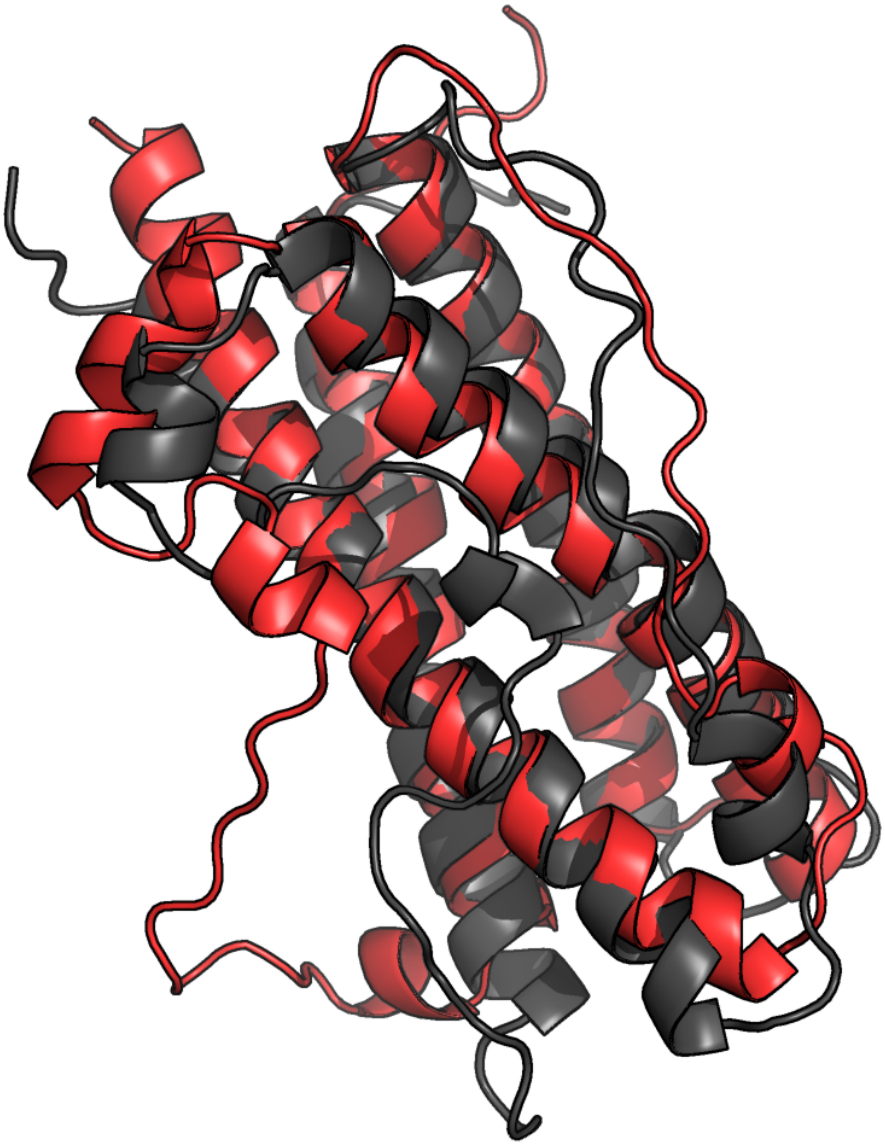
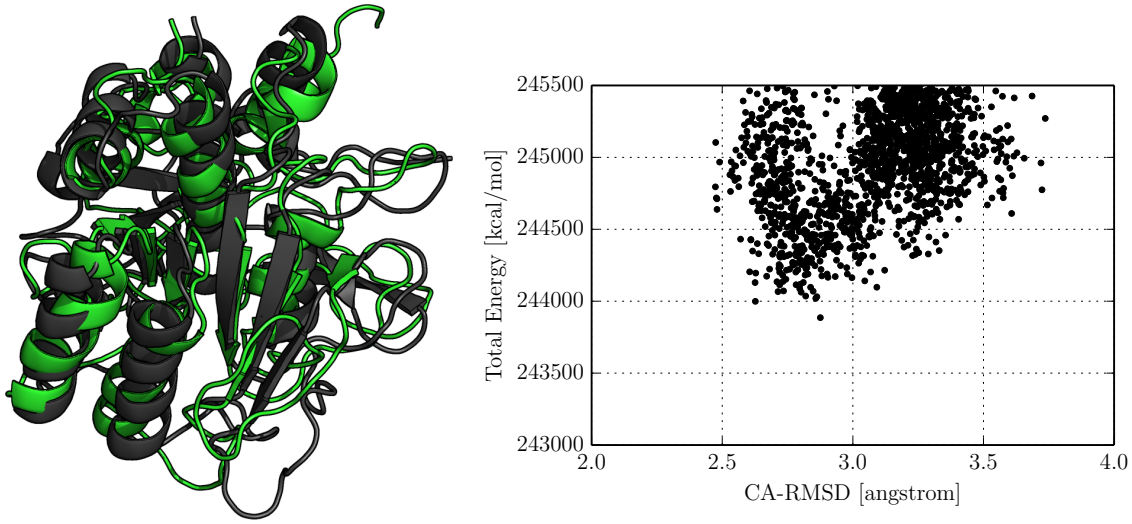


Figure 6.5: The lowest energy sample (red) for Prolactin after refinement. Note the flexible part which is not in agreement with the experimental X-ray structure (grey).



(a) Lowest RMSD sample

(b) Lowest energy sample

Figure 6.6: Refinement stage of the savinase simulation. The lowest energy sample has a CA-RMSD of 2.9 Å.

6.4 Evolutionary distance constraints

As discussed previously, it is increasingly difficult to obtain sufficient distance restraints as the size of the protein increases. A recently developed methodology uses sequence analysis to infer residue contacts in 3D space [Marks et al., 2011]. In brief, the method works by identifying sequence co-variation, which retains favorable contacts between residues. This way, pair of residues which are probable to be close in 3D space can be identified. The procedure is briefly summarized in Fig. 6.7, and is implemented in the EVfold program.

In this proof-of-concept study, 270 contacts were obtained a multiple-sequence alignment using the EVfold program (Wouter Boomsma, personal communications) for the 269 residue protein Savinase. The restraints were simply treated as NOE restraints using the old NOE code mentioned in the previous section. A similar simulation to that which folded Rhodopsin was adopted. In terms of computational resources, these were increased to 100 threads and 75×10^6 iterations, compared to only 72 threads and 50×10^6 iterations for the Rhodopsin simulation. One thread identified a native-like structure.

The folding simulation yielded a lowest energy structure around 7.5 Å CA-RMSD from native. A further refinement with the new NOE code from this structure, yielded a lowest RMSD structure at 2.9 Å CA-RMSD from the X-ray structure. The structure and an energy/RMSD plot for the refinement is shown in Fig. 6.6.

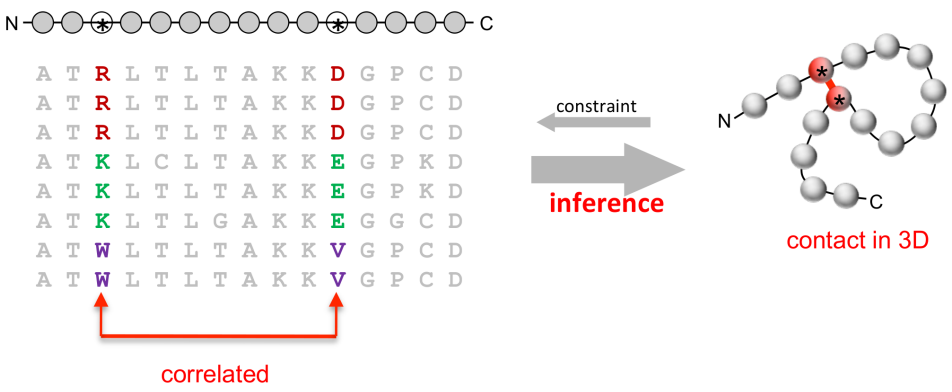


Figure 6.7: Brief overview of the process from which evolutionary constraints are inferred. Correlated sequence variation that retains favorable interactions is identified and converted to distance constraints. Figure from Marks et al., 2011.

Chapter 7

Conclusion and Outlook

During the project described in this thesis and the attached papers, I have implemented a method to determine the structure of several small proteins using their experimental chemical shifts. The structure of the CI-2 protein was solved rapidly, using only computer resources that are available in any lab, with only chemical shift data that was automatically recorded and assigned.

Lastly, I have attempted to fold several protein structures around 200 amino acids. Out of 8 proteins greater than 100 residues, a good structure was located in four cases, out of which two were larger than 200 residues. The last four likely failed due to inefficient use of the NOE restraints. Since the existing code to handle NOE restraints in PHAISTOS did not perform well on large structures, I implemented a new NOE energy term, and this was used to fold the Rhodopsin structure (225 amino acids) to a CA-RMSD of 2.5 Å from the experimental X-ray structure using a set only 195 NOE restraints and assigned backbone chemical shifts, NMR data which had been assigned through automated processes. The same code was able to fold the Savinase structure (269 amino acids) to a CA-RMSD of 2.9 Å from the experimental X-ray structure using only distance restraints derived from evolutionary data and assigned chemical shifts.

This required implementing a version of CamShift, from scratch, in PHAISTOS, and implemented useful energy function rigorously founded in Bayesian statistics. To aid the setup of calculations, a graphical user interface for PHAISTOS was created. I have parametrized and implemented a version of ProCS to calculate amide proton chemical shifts, and shown that this parametrization yields structure that are in better agreement with experimental data than simulations using a chemical shift predictor parametrized from experimental data. Furthermore, I have parametrized parts of the backbone atom ProCS chemical shift predictor and implemented this in a PHAISTOS module. This required the implementation of FragBuilder Python API which was used to automatically setup, run, and collect data from more than 2,000,000 QM calculations.

The speed of the cached version of the backbone atom ProCS chemical shift predictor will allow an energy function based on chemical shifts to be included in simulations on proteins > 200 residues. Based on results obtained on the ENHD, Protein G and CI-2 proteins, this will dramatically increase the accuracy of the energy functions that can be used to determine protein structures.

A newly developed NOE energy function shows encouraging results on folding of large structures, and further development of this module is promising.

In conclusion, I have demonstrated, that chemical shifts and sparse NOE data can, in some cases, be used with higher computational efficiency in PHAISTOS, than any other competing method. I have determined a protein structure in less than two days using automatically collected chemical shift data with computational resource available to any lab. Lastly, I have folded some of the larges protein structures ever folded using similar approaches, while using only modest amount computational resources, compared to current *state-of-the-art* methods.

Bibliography

[Bahrami et al., 2012] Bahrami, A., Tonelli, M., Sahu, S., Singarapu, K., Eghbalnia, H., and Markley, J. (2012). Robust, integrated computational control of nmr experiments to achiev e optimal assignment by adapt-nmr. *PLoS ONE*, 7:e33173.

[Berman et al., 2000] Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000). The protein data bank. *Nucl. Acids. Res.*, 28:235–242.

[Boomsma et al., 2013] Boomsma, W., Frellsen, J., Harder, T., Bottaro, S., Johansson, K. E., Tian, P., Stovgaard, K., Andreetta, C., Olsson, S., Valentin, J. B., Antonov, L. D., Christensen, A. S., Borg, M., Jensen, J. H., Lindorff-Larsen, K., Ferkinghoff-Borg, J., and Hamelryck, T. (2013). PHAISTOS: a framework for markov chain monte carlo simulation and inference of protein structure. *J. of Comp. Chem.*, 00:000–000, DOI: 10.1002/jcc.23292.

[Boomsma et al., 2008] Boomsma, W., Mardia, K., Taylor, C., Ferkinghoff-Borg, J., Krogh, A., and Hamelryck, T. (2008). A generative, probabilistic model of local protein structure. *Proc. Natl. Acad. USA*, 105:8932–8937.

[Borg et al., 2009] Borg, M., Boomsma, W., Ferkinghoff-Borg, J., Frellesen, J., Harder, T., Mardia, K. V., Rogen, P., Stovgaard, K., and Hamelryck, T. (2009). A probabilistic approach to protein structure prediction: Phaistos in casp8. *Proceedings of the 28th Leeds Annual Statistical Research Workshop, Dept. Stat., Univ. Leeds.*

[Bottaro et al., 2011] Bottaro, S., Boomsma, W., Johansson, K. E., Andreetta, C., Hamleryck, T. W., and Ferkinghoff-Borg, J. (2011). Subtle monte carlo updates in dense molecular systems. *J. Chem. Theory Comput.*, 8:695–702.

[Cavalli et al., 2007] Cavalli, A., Salvatella, X., Dobson, C. M., and Vendruscolo, M. (2007). Protein structure determination from nmr chemical shifts. *Proc. Natl. Acad. Sci.*, 104:9615–9620.

[Christensen et al., 2013] Christensen, A. S., Linnet, T. E., Borg, M., Boomsma, W., Lindorff-Larsen, K., Hamelryck, T., and Jensen, J. H. (2013). Protein structure validation and refinement using amide proton chemical shifts derived from quantum mechanics. *PLOS ONE*, page (In press).

[Christensen et al., 2011] Christensen, A. S., Sauer, S. P. A., and Jensen, J. H. (2011). Definitive benchmark study of ring current effects on amide proton chemical shifts. *J. Chem. Theory Comput.*, 7:2078–2084.

[Dolinsky et al., 2007] Dolinsky, T. J., Czodrowski, P., Li, H., Nielsen, J. E., Jensen, J. H., Klebe, G., and Baker, N. A. (2007). PDB2PQR: expanding and upgrading automated preparation of biomolecular structures for molecular simulations. *Nucl. Acids. Res.*, 35:W522–W525.

[Dolinsky et al., 2004] Dolinsky, T. J., Nielsen, J. E., McCammon, J. A., and Baker, N. A. (2004). PDB2PQR: an automated pipeline for the setup, execution, and analysis of poisson-boltzmann electrostatics calculations. *Nucl. Acids. Res.*, 32:W665–W667.

[Dunbrack and Cohen, 1997] Dunbrack, R. L. and Cohen, F. E. (1997). Bayesian statistical analysis of protein side-chain rotamer preferences. *Protein Sci.*, 6:1661–1684.

[Engh and Huber, 1991] Engh, R. A. and Huber, R. (1991). Accurate bond and angle parameters for x-ray protein structure refinement. *Acta Cryst.*, A47:392–400.

[Ferkinghoff-Borg, 2002] Ferkinghoff-Borg, J. (2002). Optimized monte carlo analysis for generalized ensembles. *Eur. Phys. J. B*, 29:481–482.

[Frisch et al., 2009] Frisch, M. J., Trucks, G. W., Schlegel, H. B., Scuseria, G. E., Robb, M. A., Cheeseman, J. R., Scalmani, G., Barone, V., Mennucci, B., Petersson, G. A., Nakatsuji, H., Caricato, M., Li, X., Hratchian, H. P., Izmaylov, A. F., Bloino, J., Zheng, G., Sonnenberg, J. L., Hada, M., Ehara, M., Toyota, K., Fukuda, R., Hasegawa, J., Ishida, M., Nakajima, T., Honda, Y., Kitao, O., Nakai, H., Vreven, T., Montgomery, Jr., J. A., Peralta, J. E., Ogliaro, F., Bearpark, M., Heyd, J. J., Brothers, E., Kudin, K. N., Staroverov, V. N., Kobayashi, R., Normand, J., Raghavachari, K., Rendell, A., Burant, J. C., Iyengar, S. S., Tomasi, J., Cossi, M., Rega, N., Millam, J. M., Klene, M., Knox, J. E., Cross, J. B., Bakken, V., Adamo, C., Jaramillo, J., Gomperts, R., Stratmann, R. E., Yazyev, O., Austin, A. J., Cammi, R., Pomelli, C., Ochterski, J. W., Martin, R. L., Morokuma, K., Zakrzewski, V. G., Voth, G. A., Salvador, P., Dannenberg, J. J., Dapprich, S., Daniels, A. D., Farkas, O., Foresman, J. B., Ortiz, J. V., Cioslowski, J., and Fox, D. J. (2009). Gaussian 09 Revision D.01. Gaussian Inc. Wallingford CT 2009.

[Harder et al., 2010] Harder, T., Boomsma, W., Paluszewski, M., Frellesen, J., Johansson, K. E., and Hamelryck, T. (2010). Beyond rotamers: a generative, probabilistic model of side chains in proteins. *BMC Bioinformatics*, 11:306–318.

[Jack and Levitt, 1978] Jack, A. and Levitt, M. (1978). Refinement of large structures by simultaneous minimization of energy and r factor. *Acta. Cryst.*, A34:931–935.

[Johansson and Hamelryck, 2013] Johansson, K. E. and Hamelryck, T. (2013). A simple probabilistic model of multibody interactions in proteins. *Proteins*, 81:1340–1350.

[Kohlhoff et al., 2009] Kohlhoff, K. J., Robustelli, P., Cavalli, A., Salvatella, X., and Vendruscolo, M. (2009). Fast and accurate predictions of protein NMR chemical shifts from interatomic distances. *J. Am. Chem. Soc.*, 131:13894–13895.

[Lange and Baker, 2012] Lange, O. F. and Baker, D. (2012). Resolution-adapted recombination of structural features significantly improves sampling in restraint-guided structure calculation. *Proteins*, 80:884–895.

[Lange et al., 2012] Lange, O. F., Rossi, P., Sgourakis, N. G., Song, Y., Lee, H.-W., Arami, J. M., Ertekin, A., Xiao, R., Acton, T. B., Montelione, G. T., and Baker, D. (2012). Determination of solution structures of proteins up to 40 kda using cs-rosetta with sparse nmr data from deuterated samples. *Proc. Natl. Acad. Sci.*, 109:10973–10878.

[Lindorff-Larsen et al., 2005] Lindorff-Larsen, K., Best, R. B., DePristo, M. A., Dobson, C. M., and Vendruscolo, M. (2005). Simultaneous determination of protein structure and dynamics. *Nature*, 433:128–132.

[Marks et al., 2011] Marks, D. S., Colwell, L. J., Sheridan, R., Hopf, T. A., Pagnani, A., Zecchina, R., and Sander, C. (2011). Protein 3d structure computed from evolutionary sequence variation. *PLoS ONE*, 6:e28766.

[Neal et al., 2003] Neal, S., Nip, A. M., Zhang, H., and Wishart, D. S. (2003). Rapid and accurate calculation of protein ^{1}h and ^{13}C and ^{15}N chemical shifts. *J. Biomol. NMR.*, 26:215–240.

[O'Boyle et al., 2011] O'Boyle, N. M., Banck, M., a C Morley, C. A. J., Vandermeersch, T., and Hutchinson, G. R. (2011). Open babel: An open chemical toolbox. *Journal of Cheminformatics*, 3:33–46.

[Oliphant, 2006] Oliphant, T. (2006). NumPy. `http://www.numpy.org/` (Accssed 10 December 2013).

[Ösapay and Case, 1991] Ösapay, K. and Case, D. A. (1991). A new analysis of proton chemical shifts in proteins. *J. Am. Chem. Soc.*, 111:9436–9444.

[Parker et al., 2006] Parker, L. L., Houk, A. R., and Jensen, J. H. (2006). Cooperative hydrogen bonding effects are key determinants of backbone amide proton chemical shifts in proteins. *J. Am. Chem. Soc.*, 128:9863–9872.

[Raman et al., 2010] Raman, S., Lange, O., Rossi, P., Tyka, M., Wang, X., Aramini, J., Liu, G., Ramelot, T., Eletsky, A., Szyperski, T., Kennedy, M. A., Prestegard, J., Montelione, G. T., and Baker, D. (2010). Rapid protein fold determination using unassigned nmr data. *Science*, 327:1014–1018.

[Rieping et al., 2005] Rieping, W., Habeck, M., and Nilges, M. (2005). Inferential structure determination. *Science*, 308:303–306.

[Robustelli et al., 2009] Robustelli, P., Cavalli, A., Dobsom, C. M., Vendruscolo, M., and Salvatella, X. (2009). Folding of small proteins by monte carlo simulations with chemical shift restraints without the use of molecular fragment replacement or structural homology. *J. Phys. Chem. B*, 113:7890–7896.

[Robustelli et al., 2010] Robustelli, P., Kohlhoff, K., Cavalli, A., and Vendruscolo, M. (2010). Using nmr chemical shifts as structural restraints in molecular dynamics simulations of proteins. *Structure*, 18:923–933.

[Rohl et al., 2004] Rohl, C. A., Strauss, C. E., Misura, K. M., and Baker, D. (2004). Protein structure prediction using rosetta methods. *Enzymol.*, 383:66–93.

[Shen and Bax, 2010] Shen, Y. and Bax, A. (2010). SPARTA+: a modest improvement in empirical NMR chemical shift prediction by means of an artificial neural network. *J. Biomol. NMR*, 48:13–22.

[Shen et al., 2008] Shen, Y., Lange, O., Delaglio, F., Aramini, J. M., Liu, G., Eletsky, A., Wu, Y., Singarapu, K. K., Lemak, A., Ignatchenk, A., Arrowsmith, C. H., Szyperski, T., Montelione, G. T., Baker, D., and Bax, A. (2008). Consistent blind protein structure generation from nmr chemical shift data. *Proc. Natl. Acad. Sci.*, 105:468–4690.

[Vila et al., 2009] Vila, J. A., Arnautova, Y. A., Martin, O. A., and Scheraga, H. A. (2009). Quantum-mechanics-derived 13Ca chemical shift server (cheshift) for protein structure validation. *Proc. Natl. Acad. Sci.*, 106:16972–16977.

[Warke and Momany, 2007] Warke, A. and Momany, C. (2007). Addressing the protein crystallization bottleneck by cocrystallization. *Cryst. Growth Des.*, 7:2219–2225.

Chapter 8

Appendix A: Published Papers

1) Definitive Benchmark Study of Ring Current Effects on Amide Proton Chemical Shifts

Anders S. Christensen, Stephan P. A. Sauer, Jan H. Jensen (2011) Definitive benchmark study of ring current effects on amide proton chemical shifts. *Journal of Chemical Theory and Computation*, 7:2078-2084.

This paper describes our computational method to calculate ring current effects on amide proton chemical shifts using QM methods. Furthermore, we compare three classical approximation to calculating ring-current effects in proteins, and find that they have similar performance. When using the parameters set obtained here, the estimated error on the classically calculated ring-current effect on amide proton chemical shifts is less than 0.1 ppm.

2) PHAISTOS: A Framework for Markov Chain Monte Carlo Simulation and Inference of Protein Structure

Wouter Boomsma, Jes Frellsen, Tim Harder, Sandro Bottaro, Kristoffer E. Johansson, Pengfei Tian, Kasper Stovgaard, Christian Andreetta, Simon Olsson, Jan B. Valentin, Lubomir D. Antonov, Anders S. Christensen, Mikael Borg, Jan H. Jensen, Kresten Lindorff-Larsen, Jesper Ferkinghoff-Borg, Thomas Hamelryck (2013) PHAISTOS: A framework for Markov chain Monte Carlo simulation and inference of protein structure. *Journal of Computational Chemistry*, 34:1697-1705.

In this paper, we describe the PHAISTOS program.

3) Protein Structure Validation and Refinement Using Amide Proton Chemical Shifts Derived from Quantum Mechanics

Anders S. Christensen, Troels E. Linnet, Mikael Borg, Wouter Boomsma, Kresten Lindorff-Larsen, Thomas Hamelryck, Jan H. Jensen (2013) Protein Structure Validation and Refinement Using Amide Proton Chemical Shifts Derived from Quantum Mechanics. *PLoS ONE* 8:e84123.

In this paper, we demonstrate the accuracy of the QM-derive amide proton chemical shift predictor called ProCS.

4) FragBuilder: An efficient Python library to setup quantum chemistry calculations on peptides models

Anders S. Christensen, Thomas Hamelryck, Jan H. Jensen (2014) FragBuilder: An efficient Python library to setup quantum chemistry calculations on peptides models. *PeerJ* 2:e277.

This paper presents the FragBuilder Python API, which we developed to set up calculation of nearly 2,000,000 QM calculation in the development of ProCS.

Definitive Benchmark Study of Ring Current Effects on Amide Proton Chemical Shifts

Anders S. Christensen, Stephan P. A. Sauer, Jan H. Jensen (2011) Definitive benchmark study of ring current effects on amide proton chemical shifts. *Journal of Chemical Theory and Computation*, 7:2078-2084.

JCTC Journal of Chemical Theory and Computation

ARTICLE

pubs.acs.org/JCTC

Definitive Benchmark Study of Ring Current Effects on Amide Proton Chemical Shifts

Anders S. Christensen,* Stephan P. A. Sauer, and Jan H. Jensen*

Department of Chemistry, University of Copenhagen, Universitetsparken 5, DK-2100 Copenhagen Ø, Denmark

Supporting Information

ABSTRACT: The ring current effect on chemical shifts of amide protons ($\Delta\delta_{RC}$) is computed at the B3LYP/6-311++G(d,p)//B3LYP/aug-cc-pVTZ level of theory for 932 geometries of dimers of *N*-methylacetamide and aromatic amino acid side chains extracted from 21 different proteins. These $\Delta\delta_{RC}$ values are scaled by 1.074, based on MP2/cc-pVQZ//B3LYP/aug-cc-pVTZ chemical shift calculations on four representative formamide/benzene dimers, and are judged to be accurate to within 0.1 ppm based on CCSD(T)/CBS//B3LYP/aug-cc-pVTZ calculations on formamide. The 932 scaled $\Delta\delta_{RC}$ values are used to benchmark three empirical ring current models, including the Haigh−Mallion model used in the SPARTA, SHIFTX, and SHIFTS chemical shift prediction codes. Though the RMSDs for these three models are below 0.1 ppm, deviations up to 0.29 ppm are found, but these can be decreased to below 0.1 ppm by changing a single parameter. The simple point-dipole model is found to perform just as well as the more complicated Haigh−Mallion and Johnson−Bovey models.

1. INTRODUCTION

Prediction of chemical shifts in proteins based on protein structure serves many uses in structure verification or fast generation of structures in accordance with relatively inexpensive experimental nuclear magnetic resonance (NMR) data.[1−5] The most popular protein chemical prediction software packages include the SHIFTX,[6] SHIFTS,[7,8] SPARTA,[9] and PROSHIFT[10] servers. These programs use empirical models that relate various features of protein structure, such as secondary structure, hydrogen bond geometry, and ring current effects, to changes in chemical shifts. The contributions from several different sources of chemical shift perturbations are in all cases assumed to work additively. These small additive terms are in many cases given as classical approximations to well-known physical interactions. For example, SPARTA, SHIFTX, and SHIFTS use the approximation due to Haigh and Mallion[11] to model the changes in chemical shifts due to ring current effects in the aromatic side chains of phenylalanine, tyrosine, tryptophan, and histidine residues. The Haigh−Mallion model contains a single adjustable parameter for each side chain, which must be parametrized from a data set. In the case of SHIFTX, these parameters are obtained by a data mining approach, which correlates experimental chemical shifts with corresponding, known X-ray protein structures, based on a series of predefined physical and empirical terms.[6] In the case of SHIFTS, the parameters are obtained by fitting parameters for a set of known physical terms, which relates protein structure and chemical shifts to a data set which combines empirical chemical shift data as well as data obtained through quantum chemical calculations.[12] Ultimately, these fitting methods include uncertainties from many terms in the underlying approximations of the fit, as well as the uncertainties connected to the experimental data. It is thus unclear how accurate the obtained parameters are in reproducing the underlying physics of the system.[13]

Other methods exist to approximate the ring current effect, most notably[11,14] the Johnson−Bovey model[15] and the simpler point-dipole model due to Pople.[16,17] In general, the three methods describe the change in chemical shift due to a nearby aromatic ring, formally as

$$\Delta\delta_{RC} = i\,B\,G \tag{1}$$

where G is a geometric factor, which depends on the spatial orientation and distance of the ring relative to the proton, B is a "natural constant" denoting the ring current intensity for a benzene ring, and i is the ring current intensity relative to that of a benzene ring, such that $i_{benzene} \equiv 1$. A thorough description of the three models mentioned above can be found in the Supporting Information. It has previously been attempted to derive analytical expressions for the values of i and B for different functional forms of G, but these have not been successful in reproducing experimental results.[11] Consequently, various numerical methods have been widely used to obtain the intensity values.

The study by Case[18] is one of the few that has addressed these issues using *ab initio* calculations. A methane molecule is used to probe the chemical shift perturbation due to a nearby aromatic ring, with the chemical shift calculated at the CSGT/PW91/IGLO-III level of theory. However, it is not clear whether the level of theory used at that time (1995) is sufficiently accurate, and second, it is unknown whether the ring current parameters obtained for a methane proton can directly be transferred to amide protons. The parameters obtained by Case are used in the SPARTA program.[9]

In this study, we use CCSD(T) and MP2 methods to benchmark the accuracy of $\Delta\delta_{RC}$ calculations at the B3LYP/6-311++G(d,p)//B3LYP/aug-cc-pVTZ level of theory. This level of theory is then used to compute nearly 1000 representative

Received: April 15, 2011
Published: June 01, 2011

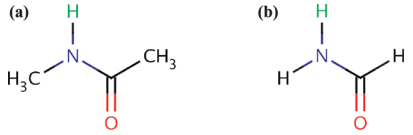


Figure 1. The molecules used as probes for the ring current effects on amide protons: *trans*-*N*-methylacetamid (NMA) (a) and formamide (FMA) (b). The probe nucleus for which the shielding constants are calculated are the amide proton of NMA and the amide proton *trans* to the C═O bond in FMA, here marked in green color.

$\Delta\delta_{RC}$ values, which, in turn, are used to obtain parameters for three empirical $\Delta\delta_{RC}$ models.

The paper is organized as follows. First, we describe the computational methodology used to isolate the ring current effect and to obtain data sets, against which the ring current intensity parameters are fitted. Then, we benchmark various levels of theory, in order to estimate error bounds on our data. This is followed by a presentation of the obtained intensities and a comparison to the intensities obtained by other authors.

2. COMPUTATIONAL METHODOLOGY

2.1. Isolating the Ring Current Effect. As a probe nucleus, for which the isotropic shielding is calculated using quantum chemical methods, the amide proton of a *trans*-*N*-methylacetylamide (NMA; Figure 1a) molecule is used in order to provide a small and inflexible molecule with a high degree of resemblance to the amide group of the protein backbone. For more computationally demanding calculations, such as MP2, CCSD, and CCSD(T) calculations, the two methyl groups are removed in order to save computational time, and the probe nucleus is then the amide proton *trans* to the C═O bond in formamide (FMA; Figure 1b).

A large number of dimer systems (see next subsection), consisting of an amide probe molecule and a nearby aromatic ring in different conformations is constructed. Following the general approach of Boyd and Skrynnikov,[19] the absolute shielding of the probe hydrogen in the dimer system can be written as

$$\sigma_H^{Dimer} = \Delta\sigma_{Conformation} + \Delta\sigma_{Local} + \Delta\sigma_{RC} \quad (2)$$

The chemical shift of the probe hydrogen atom in the dimer system will, apart from the ring current effect, also be influenced by the exact geometry and type of the probe molecule (described in the $\Delta\sigma_{Conformation}$ term) as well as any possible interactions with the aromatic moiety, such as electrostatic forces, possible hydrogen bonding, spin−spin repulsion, and other effects which can be difficult to quantify and separate (described in the $\Delta\sigma_{Local}$ term). Finally, the chemical shift perturbation due to the aromatic ring is approximated as $\Delta\sigma_{RC}$.

Reference systems for each dimer system, which have approximately identical local interactions between the molecules, apart from the ring current effect, are constructed in order to filter out these hard-to-quantify effects (see Figure 2). These are modeled as corresponding dimer systems, where the aromatic ring has been replaced by an olefinic analogue. The definition of an olefinic analogue here is an aromatic ring which, by the addition of two hydrogen atoms, has lost its aromaticity. The protonation is done such that the planar geometry of the ring is still enforced, causing deviations in the spatial positions relative to the corresponding aromatic ring dimer to be negligible. The olefinic

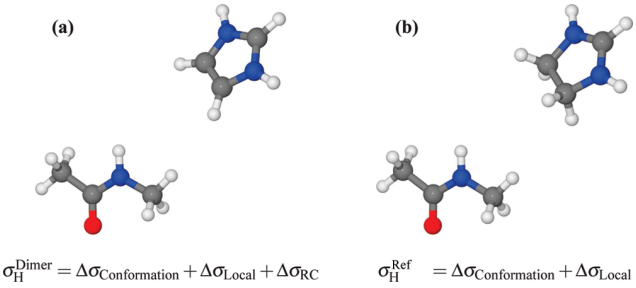


Figure 2. Two example geometries demonstrating the two different dimers used in the calculation scheme to isolate the ring current effect for one amide-ring conformation. The shown geometries correspond to the amide proton of ILE64 and the side chain of HIS69 in the HIV-1 protease, PDB-code 2I4V. In part a, the chemical shift of the probe nucleus is determined by the conformation of the NMA molecule, electrostatic and spin−spin repulsion interactions to the positively charged imidazolium molecule, and, by comparison, a small ring current interaction. In part b, the aromaticity of the imidazolium is broken, but the spatial distribution of charge as well as the internal conformation of NMA is approximately identical to those found in part a.

analogue is placed such that the ring center corresponds to the center of the aromatic ring, relative to the probe hydrogen, and the coordinates of the carbon/nitrogen atoms are matched as closely as possible. This approach ensures that $\Delta\sigma_{Conformation}$ and $\Delta\sigma_{Local}$ are largely retained, while $\Delta\sigma_{RC}$ is removed. Using this substitution scheme, the absolute shielding of the hydrogen atom in the reference system can be written as

$$\sigma_H^{Ref} = \Delta\sigma_{Conformation} + \Delta\sigma_{Local} \quad (3)$$

which enables us to estimate the ring current contribution to the chemical shift due to the aromatic ring as

$$\Delta\delta_{RC} = -\Delta\sigma_{RC} \approx \sigma_H^{Ref} - \sigma_H^{Dimer} \quad (4)$$

The aromatic rings studied here are equivalent to the rings found in the aromatic protein side chains. See Table 1 for an overview of the used molecules. Sketches are shown in the Supporting Information as well.

2.2. Construction of Test Systems. Dimers consisting of an amide probe and an aromatic ring were generated from a data set of 21 protein structures obtained from the RCSB Protein Data Bank (PDB),[20] in order to to ensure that only realistic conformations were used in the QM calculations.

The structures used were (listed by PDB code): 1F94, 1GK1, 1IGD, 1JYQ, 1JYR, 1JYU, 1Q3E, 1QJP, 1VJC, 1XA5, 1ZJK, 2ACO, 2B6C, 2D57, 2DRJ, 2ETL, 2F47, 2FZG, 2GOL, 2I4D, and 2I4V. Since the used structures were experimental X-ray structures, no hydrogen atoms were present in the structures, and PDB2PQR 1.5[21,22] was used to protonate the structures in order to obtain hydrogen atom positions. From all of these protein structures, we selected systems where the center of an aromatic ring was within a cutoff distance of 7 Å from an amide proton. This resulted in a total of 932 different dimer conformations (see Table 1). For each of these conformations, a dimer was created with a simpler aromatic ring in place of the aromatic side chain with the ring centers at identical coordinates and in the same plane. A directional vector was used to align the rotation of the ring in the plane, in order to have closely matching coordinates for the heavy atoms. For tyrosine, the center to oxygen vector was used. For benzene, the center to C^{γ} to C^{ζ} vector was used.

Table 1. List of the Side Chain Approximations Used in This Work and Their Olefinic Analogues[a]

side chain	analogue	olefinc analogue	# dimers
phenylalanine	benzene	1,4-cyclohexadiene	276
tyrosine	phenol	cyclohexa-1,4-diene-1-ol	172
tryptophan	indole	2,3,5,6-tetrahydroindole	113
histidine	imidazole	4,5-dihydroimidazole	185
histidine$^+$	imidazolium	4,5-dihydroimidazolium	174

[a] The residue type is listed along with the aromatic and olefinic analogues, as well as the total number of different NMA/ring dimer conformations in each data set.

For histidine (both in the charged and neutral state), the center to $C^{\varepsilon 1}$ vector was used. The same histidine−amide group pairs were used to generate the dimers for both charged and neutral histidine conformations. The $N^{\varepsilon 2}$ nitrogen atom was in all cases of neutral histidine assumed to be the deprotonated nitrogen.

A given backbone amide group within the 7 Å range from the aromatic ring was substituted by an NMA or an FMA molecule, with the nitrogen atoms at identical coordinates. Furthermore, the N−H vector and the C(=O)−N−H plane were also aligned. See Table 1 for the number of dimers for each ring type.

If the aromatic ring corresponded to the ring of the side chain of the previous residue, with respect to the investigated amide group, the dimer construction scheme occasionally caused a clash between the extra hydrogens, where the C^{β} atom was previously located. Other conformations also gave rise to unphysical conformations, due to clashes between the inflexible subunits of the constructed dimers. To avoid computational artifacts from these, all dimers with a shortest intermolecular distance of 3.4 Å or less were discarded, since 3.4 Å is twice the van der Waal radius of the largest atom (carbon) in the system.[23] NMR shielding constants were then calculated for the dimer systems.

2.3. Basis Set Extrapolations and Correlation Effects. In this work, density functional theory (DFT; and the very popular B3LYP functional[24,25]) is used to obtain NMR shielding constants. Due to the partly semiempirical nature of the approximated exchange-correlation functionals used, B3LYP data cannot in general be expected to show convergence toward experimental shielding values or values obtained at very accurate levels of theory when increasing the basis set size.[26] It is, however, often the case that a small error can be obtained in calculated DFT chemical shielding constants, compared to high-level correlated wave function methods, if a simple linear correction or scaling factor is applied to the DFT data.[27] In this work, a comparison of B3LYP to high-level correlated methods is used to obtain such a linear scaling factor.

Unfortunately, CCSD(T) calculations with an appropriate basis set are still not possible for the FMA/benzene dimer, which has 70 electrons. Instead, we benchmark the shielding constants for FMA alone at the B3LYP, MP2, CCSD, and CCSD(T) levels of theory, in order to allow us to estimate error bounds to shielding constants obtained at levels of theory less accurate than CCSD(T).

For complete basis set limit (CBS) estimates, we use the approach of Moon and Case[26] and Kupka et al.[27,28] By using Dunning's correlation consistent basis sets[29] (cc-pVxZ; where $x \in \{D, T, Q, 5, 6, ...\}$ is the valence orbital splitting in the basis set), it is possible to carry out calculations using a sequence of basis sets of well-defined, systematic increasing quality. Kupka et al. extrapolate calculated NMR shielding values toward infinite basis set size with a three parameter exponential decreasing function:

$$\sigma(x) = \sigma(\infty) + A \exp(-x/B) \quad (5)$$

where $\sigma(x)$ is the shielding obtained using a basis set with the valence orbital splitting number of x and $\sigma(\infty)$, A and B are the fitting parameters, with $\sigma(\infty)$ being the estimated shielding in the complete basis limit. A nonlinear least-squares Marquardt−Levenberg algorithm[30,31] is used to fit the parameters.

Jensen has constructed a set of basis sets for the purpose of Hartree−Fock (HF) and DFT NMR shielding calculations, called the polarization consistent pcS-n basis sets.[32] For basis sets of similar valence orbital splitting, the pcS-n basis sets contain more basis functions of low angular momentum, compared to the Dunning-type basis sets. pcS-1 is a double-ζ quality basis set, pcS-2 is triple-ζ, and so forth. When estimating the complete basis limit based on the pcS-n basis sets, a value of $x = n + 1$ is thus used in eq 5.

Last, we compare the proton chemical shift of the amide proton trans to the C=O bond in FMA at the CCSD(T)/CBS level of theory to the experimental value, in order to verify that CCSD(T)/CBS is, in fact, a reliable method. Inferring the experimental gas-phase ^{1}H shielding values from CH_4 (σ_H = 30.61 ppm[33]), an experimental value of σ_H = 26.24 ppm in the gas phase[34] at 483 K is obtained. At this temperature, thermal motion cause rapid switching of the two N-amide protons and the peaks are not separable, so this value has to be considered as an average over the two proton chemical shifts.[34]

It is well-known[35−38] that a zero-point vibrational correction (ZPVC) has to be added to equilibrium geometry *ab initio* shielding constants in order to obtain close agreement to experimental data. This vibrational averaging correction can easily be calculated using the method of Kern and Matcha.[39] While we had preferred to carry out a ZPVC calculation at the same level of theory as the geometry optimization of the molecules used throughout this work, no program is currently capable of automatically computing a ZPVC at the DFT level of theory with Gaussian-type basis sets. In the following, the ZPVC is calculated at the MP2/cc-pVQZ level of theory instead.

2.4. Software. All geometries were minimized at the B3LYP/aug-cc-pVTZ level of theory using Gaussian 03,[40] except when otherwise noted. All DFT calculations of NMR shielding constants were carried out using Gaussian 03. MP2/6-311++G(d,p) NMR calculations were also carried out in Gaussian 03, while the calculation of MP2 shielding constants using Dunning's correlation consistent basis sets[29] and the polarization consistent pcS-n and aug-pcS-n basis sets of Jensen[32] were carried out with Turbomole 6.2.[41] All calculations at the CCSD and CCSD(T) levels of theory were carried out using CFOUR 1.0.[42] All NMR shielding constants are calculated using the Gauge-Including Atomic Orbital formulation.[43−45] For the calculations of the ZVPC to the FMA chemical shifts, the equilibrium geometry of a planar FMA molecule was obtained at the MP2/cc-pVQZ level of theory with CFOUR 1.0, exploiting the C_S symmetry of the molecule. From this equilibrium geometry, the ZVPC to the NMR isotropic shielding was subsequently calculated at the MP2/cc-pVQZ level of theory using the method of Kern and Matcha[39] as implemented in CFOUR 1.0.

Table 2. Absolute Isotropic Chemical Shielding of the *cis*-*N*-amide Proton in Gas-Phase FMA at the B3LYP, MP2, CCSD, and CCSD(T) Levels of Theory Using Dunning's Correlation Consistent Basis Sets and the Polarization Consistent Shielding Basis Sets of Jensen[a]

		method			
basis set	size	CCSD(T)	CCSD	MP2	B3LYP
cc-pVDZ	57	28.06	28.09	27.90	27.67
cc-pVTZ	132	27.29	27.35	27.16	27.17
cc-pVQZ	255	26.92	27.00	26.80	26.94
cc-pV5Z	438	26.78	26.86	26.65	26.83
$\sigma_{\text{cc-pVxZ}}(\infty)$		26.64	26.73	26.50	26.73
pcS-0	44	29.32	29.36	29.31	28.88
pcS-1	66	27.55	27.58	27.40	27.29
pcS-2	141	27.02	27.09	26.89	26.91
pcS-3	321	26.75	26.83	26.62	26.77
$\sigma_{\text{pcS-}n}(\infty)$		26.67	26.78	26.57	26.75
$\sigma_{\text{exptl(gas)}}$			26.24		

[a] All values are given as ppm. The experimental value is obtained at 483 K.[34] $\sigma(\infty)$ is obtained using eq 5 and fitted over all values in the corresponding series of basis sets. The size indicates the number of basis functions in the system at the given basis set size. All shielding constants are given in ppm.

3. RESULTS

3.1. Correlation and Basis Set Effects on the Chemical Shift of the (N−)H Proton in Formamide. It is currently not feasible to perform a complete basis set study at the CCSD(T) level for $\Delta\delta_{\text{RC}}$ of a FMA/benzene dimer. Instead, we perform such a study of the chemical shielding of the *cis*-*N*-proton in FMA and use the results to benchmark more approximate methods that can be applied to FMA/benzene dimers (as described in the next subsection).

Table 2 lists CCSD(T) chemical shielding values computed using a B3LYP/aug-cc-pVTZ optimized geometry of FMA and two different, systematic series of basis sets (cc-pV*x*Z and pcS-*n*). Each set of calculations is used to extrapolate shielding constants to the complete basis set limit (as described in the previous section) and lead to very similar results: 26.64 and 26.67 ppm for cc-pV*x*Z and pcS-*n*, respectively. In the following, we will refer to 26.64 ppm as CCSD(T)/CBS, since this value is extrapolated using the largest basis set (cc-pV5Z) and since the Dunning-type basis sets are constructed for the purpose of correlated wave function calculations, whereas the pcS-*n* basis sets are constructed specifically for shielding constant calculations at the HF and DFT levels of theory.

The CCSD(T)/CBS value is 0.40 ppm higher than the experimental gas phase value obtained at 483 K of 26.24 ppm. However, this experimental value includes vibrational effects and is an average of the chemical shieldings of both amide protons. The effect of vibrations at 0 K (i.e., the zero-point vibrational correction) can be estimated relatively easily, as described in the previous section. At the MP2/cc-pVQZ level of theory, the ZPVC is −0.26 ppm, which, when used to correct the CCSD(T)/CBS value, results in a chemical shielding of 26.38 ppm—within 0.14 ppm of experiment. The ZPVC correction is unlikely to contribute significantly to $\Delta\delta_{\text{RC}}$, because it is a shielding difference between two molecular systems with very similar vibrational normal modes of FMA. Thus, we in the following focus on the electronic contribution to the chemical shielding alone.

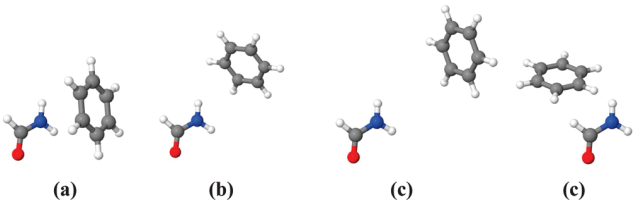


Figure 3. Pictures of the four FMA/benzene dimers used in this study. The resulting $\Delta\delta_{\text{RC}}$ calculated at various levels of theory for each conformation can be found in Table 3, where $\Delta\delta^1_{\text{RC}}$ corresponds to conformation a, $\Delta\delta^2_{\text{RC}}$ to conformation b, and so forth.

The CBS values computed using CCSD, MP2, and B3LYP are all within 0.14 ppm of the CCSD(T)/CBS value, suggesting that the amide proton chemical shielding is relatively insensitive to electronic correlation effects in the limit of large basis sets. However, it is quite basis-set-dependent as at least the cc-pV5Z or the pcS-3 basis set is needed to get within 0.2 ppm of the CCSD(T)/CBS, with the exception of MP2/cc-pVQZ, which deviates by 0.16 ppm. We therefore choose MP2/cc-pVQZ for the $\Delta\delta_{\text{RC}}$ calculations using the FMA/benzene dimers described in the next subsection. Since $\Delta\delta_{\text{RC}}$ is a relative shielding value between two very similar systems, we expect that the MP2/cc-pVQZ results are well within 0.1 ppm of what would be computed with CCSD(T)/CBS and measured experimentally. A factor not investigated here was the dependence on the used geometry, which is known to cause deviations in calculated 1H shielding constants on the order of ±0.1 ppm—see for instance Rablen et al.[46]

3.2. Scaling B3LYP Results to Those Obtained with Correlated Wave Function Methods. In this section, high-level correlated wave function methods are used to obtain a linear scaling correction to the chemical shift contribution due to ring current effects, obtained at the B3LYP/6-311++G(d,p)//B3LYP/aug-cc-pVTZ level of theory.

Four dimer systems were selected from the large data set of NMA/benzene dimers (see Figure 3), in such a way that the ring current contributions ($\Delta\delta_{\text{RC}}$) in the four dimer conformations cover a range from −0.72 ppm to +0.15 ppm, at the B3LYP/6-311++G(d,p) level of theory, in even sized steps. In these dimers, the NMA molecule was replaced with the much smaller FMA molecule, and the isotropic shielding was calculated using various methods and basis sets. Here, the chemical shift is modeled by

$$\Delta\delta_{\text{RC}}^{(\text{uncorrected})} \approx \sigma_{\text{H}}^{\text{Probe}} - \sigma_{\text{H}}^{\text{Dimer}} \qquad (6)$$

where $\sigma_{\text{H}}^{\text{Probe}}$ is the shielding of the probe nucleus in the probe molecule alone and $\sigma_{\text{H}}^{\text{Dimer}}$ is the shielding of the probe nucleus in the probe molecule in the dimer. Note that an NMR calculation for a reference dimer is not carried out, and the linear scaling factor is unaffected, whether a reference calculation is carried out, since this calculation would also have to be scaled by the same factor. The results are collected in Table 3. We observe the following:

1. Regardless of the basis set or method, the obtained $\Delta\delta_{\text{RC}}$'s have a linear correlation to B3LYP/6-311++G(d,p) data of 0.992 or better (see the Supporting Information). It is thus demonstrated that applying a linear correction based

Table 3. Shielding Constant of the FMA Probe Proton in a Vacuum for Each Method and Basis Set Used in This Section, As Well As the Chemical Shift Ring Current Interaction ($\Delta\delta^{n}_{RC}$) for Each of the Four Different Conformations Used[a]

method	σ_{FMA}	$\Delta\delta^{1}_{RC}$	$\Delta\delta^{2}_{RC}$	$\Delta\delta^{3}_{RC}$	$\Delta\delta^{4}_{RC}$	scaling
B3LYP/6-311++G(d,p)	27.64	−0.72	−0.43	−0.21	0.15	
MP2/6-311++G(d,p)	27.70	−0.76	−0.45	−0.22	0.15	1.052
CCSD/6-311++G(d,p)	27.90	−0.75	−0.44	−0.22	0.15	1.033
CCSD(T)/6-311++G(d,p)	27.85	−0.73	−0.43	−0.21	0.15	1.012
MP2/cc-pVDZ	27.90	−0.74	−0.43	−0.22	0.17	1.033
MP2/cc-pVTZ	27.16	−0.77	−0.46	−0.23	0.17	1.076
MP2/cc-pVQZ	26.80	−0.81	−0.45	−0.22	0.13	1.074
B3LYP/pcS-0	28.88	−0.75	−0.42	−0.21	0.16	1.042
B3LYP/pcS-1	27.29	−0.76	−0.44	−0.22	0.16	1.056
B3LYP/pcS-2	26.91	−0.80	−0.47	−0.24	0.14	1.087
B3LYP/pcS-3	26.77	−0.80	−0.47	−0.24	0.16	1.097
B3LYP/pcS-4	26.76	−0.80	−0.47	−0.25	0.16	1.095
CCSD(T)/CBS	26.64					
B3LYP/6-311++G(d,p) (NMA)		−0.74	−0.43	−0.22	0.14	1.004

[a] Furthermore, the resulting scaling factor relative to data obtained at the B3LYP/6-311++G(d,p) level of theory is noted. The B3LYP/6-311++G(d,p) shieldings for the identical conformation with NMA as a probe are given in the bottom row. All shielding constants and $\Delta\delta_{RC}$ values are given in ppm.

on correlated methods is a very good approximation. No constant offset (intercept) greater that 0.01 ppm was found, so the relationship between data obtained at the B3LYP level of theory and data obtained using a correlated wave function method is effectively a simple scaling factor. Thus, the fit was carried out as

$$\Delta\delta^{Other}_{RC} = k_{scaling}\Delta\delta^{B3LYP}_{RC} \quad (7)$$

where $\Delta\delta^{B3LYP}_{RC}$ is the $\Delta\delta_{RC}$ obtained at the B3LYP/6-311++G(d,p) level of theory, $\Delta\delta^{Other}_{RC}$ is the $\Delta\delta_{RC}$ obtained using another method and/or basis set, and $k_{scaling}$ is the fitted scaling constant. Here, the $\Delta\delta_{RC}$ values are obtained via eq 6.

2. All $\Delta\delta_{RC}$ values are within 0.1 ppm of one another, including B3LYP/6-311++G(d,p), which is used for the 932 dimer calculations. This supports our previous assertion that it is easier to compute $\Delta\delta_{RC}$ accurately compared to the computation of absolute shielding constants. Therefore, the $\Delta\delta_{RC}$ values listed in Table 3 are very likely within 0.1 ppm of what would be computed with CCST(T)/CBS and measured experimentally.
3. As a result, all scaling factors are within 10% and will all yield very similar results. However, on the basis of the results in Table 2, we pick the scaling factor computed at the MP2/cc-pVQZ level, where $k_{scaling}$ = 1.074.
4. The difference between ring current effects acting on either an FMA or an NMA probe was found at the B3LYP/6-311++G(d,p) level of theory to be a factor of 1.004 (see Table 3). This suggests that results obtained using FMA as a probe are, to a very good approximation, transferable to systems where NMA is used as a probe.

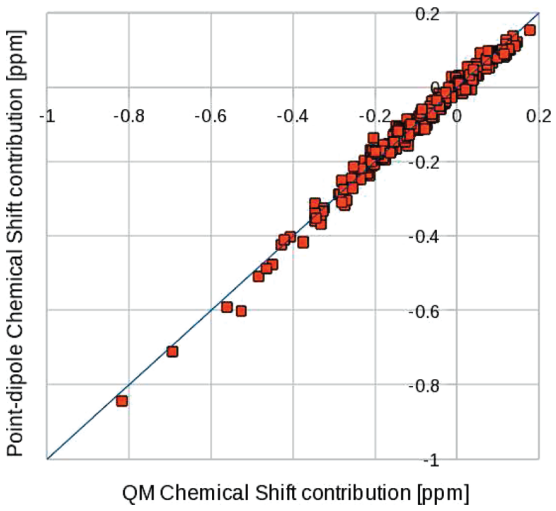


Figure 4. Correlation between the chemical shift predictions of the point-dipole model and the chemical shifts obtained by eq 4 for a set of NMA/benzene dimers using a best fit value of B_{PD} = 30.42 ± 0.16 ppm Å^3. The blue line represents the best fit between the two methods. The linear correlation of the data set is 0.993.

3.3. Expressions for the *B* Factors. In the point-dipole model, the definition of $i_{Benzene} \equiv 1$ is used, and the B factor in the point-dipole model (B_{PD}) is obtained via a fit using the chemical shifts obtained for all NMA/benzene dimers to their corresponding G values in the point-dipole model ($G_{PD}(\vec{r},\theta)$, see Supporting Information), using the following formula:

$$\Delta\delta_{RC} = B_{PD}G_{PD}(\vec{r},\theta) \quad (8)$$

where $\Delta\delta^{QM}$ is the scaled QM calculated chemical shifts of the amide protons using eq 4 and $G_{PD}(\vec{r},\theta)$ is the geometric term of the NMA/benzene dimers in the point-dipole model. This gives a value of B_{PD} = 30.42 ± 0.16 ppm Å^3. The linear correlation of this fit is r = 0.993. See Figure 4 for a scatter plot of the fitted data set.

In the literature, the trend has been to use formally derived B factors in the Johnson−Bovey model (B_{JB}) and scale the relative intensities accordingly.[18] Following this, the analytical values of B_{JB} are used in the Johnson−Bovey model in this work. These evaluate to −3.79 ppm and −3.25 ppm for five- and six-membered rings, respectively. To facilitate an easy comparison of the ring current intensities to those found by Case,[18] a B factor in the Haigh−Mallion model of B_{HM} = 5.455 ppm Å is adopted.

3.4. Fitting the Relative *i* Factors. Using the B factor obtained in the previous subsection, the relative ring current intensities of all ring types in the three ring current models are obtained as the best fit of i when fitting the right-hand side of eq 4 to the right-hand side of eq 1.

The relative intensities of the two rings in tryptophan were trivially fitted using a two parameter fitting routine, although the contributions from the five- and six-membered rings were somewhat correlated. The fitted relative ring current intensities can be found in Table 4, which also features a comparison to the i factors found in other studies. A comparison is made to the values used in the SHIFTX and SHIFTS programs and to the values obtained for methane hydrogen by Case[18] as used in SPARTA. The linear correlation between the B3LYP/6-311++G(d,p) ring current contributions and the predictions of the three approximations

Table 4. Relative Ring Current Intensity Factors of the Different Side Chains, As Found in This Study, Compared to the Value of Other Studies[a]

model	point-dipole	Haigh−Mallion				Johnson−Bovey	
reference	This Work	SHIFTX[6]	SHIFTS[47]	Case[18] SPARTA[9]	This Work	Case[18]	This Work
PHE	1.00 (0.02, 0.07)	1.05 (0.05, 0.18)	1.00 (0.05, 0.17)	1.46 (0.07, 0.17)	1.18 (0.03, 0.06)	1.27 (0.03, 0.14)	1.13 (0.02, 0.06)
TYR	0.81 (0.02, 0.10)	0.92 (0.02, 0.09)	0.84 (0.02, 0.08)	1.24 (0.06, 0.22)	0.93 (0.02, 0.09)	1.10 (0.04, 0.10)	0.91 (0.02, 0.07)
HIS+	0.69 (0.02, 0.05)	0.43 (0.08, 0.29)	0.90 (0.06, 0.12)	1.35 (0.05, 0.07)	1.26 (0.03, 0.05)	1.40 (0.03, 0.08)	1.27 (0.03, 0.05)
HIS	0.68 (0.03, 0.06)	0.43 (0.08, 0.28)	0.90 (0.07, 0.11)	1.35 (0.06, 0.08)	1.22 (0.03, 0.07)	1.40 (0.04, 0.09)	1.25 (0.03, 0.06)
TRP5	0.57 (0.03, 0.08)	0.90 (0.03, 0.11)	1.04 (0.03, 0.08)	1.32 (0.04, 0.15)	0.97 (0.02, 0.10)	1.02 (0.02, 0.10)	1.06 (0.02, 0.09)
TRP6	1.02	1.04	1.02	1.24	1.18	1.27	1.18
B-factor	30.42 ppm Å^3	5.13 ppm Å	5.455 ppm Å	5.455 ppm Å	5.455 ppm Å	−3.25 ppm[b]	−3.25 ppm[b]
						−3.79 ppm[b]	−3.79 ppm[b]

[a] The RMSD associated with using the given intensity factor and *B* value is given as the first entry in the parentheses, and MAD as the second entry, for each intensity factor. The RMSD and MAD are calculated over all dimer systems used in the fits to obtain intensity factors. The RMSD given for tryptophan is the RMSD for a sum of both rings with the given intensities. [b] In the Johnson−Bovey model, values of −3.25 ppm and −3.79 ppm are used for six- and five-membered rings, respectively.

were found to be $r = 0.980$ or better (see Supporting Information), so using linear fits to determine the *i* factors is evidently a very good approximation. For the data sets for each ring type, chemical shift predictions of different sets of *i* factors are compared to our QM data. We present the *root-mean-square deviation* (RMSD) and the *maximum absolute deviation* (MAD) of the data set. The RMSDs to our QM values are seemingly very small for all sets of intensity parameters. However, this is mostly due to the magnitude of the ring current effect in the dimers used in the data set being on average very small, and RMSD is thus not a very good measure in this case. Our intensities, however, do have the lowest maximum RMSD of up to just 0.03 ppm, while the competing methods have RMSDs of up to 0.08 ppm (SHIFTX), 0.07 ppm (SHIFTS), 0.07 ppm (Case, Haigh−Mallion), and 0.04 ppm (Case, Johnson−Bovey) for a residue type. A much better metric than RMSD is in this case the maximum average deviation (MAD) to QM values, which loosely corresponds to the largest error one can expect from using a certain set of intensities. In this metric, our method has a MAD of 0.05−0.10 ppm or better, while the corresponding numbers for the competing methods are 0.09−0.29 ppm (SHIFTX), 0.08−0.17 ppm (SHIFTS), 0.09−0.29 ppm (Case, Haigh−Mallion), and 0.09−0.14 ppm (Case, Johnson−Bovey), for all residue types.

We note that the SHIFTX and SHIFTS predictions are, on average, slightly lower than our prediction and those of Case. One possible explanation is that the SHIFTX and SHIFTS predictions are based on empirical parameters fitted to chemical shifts measured for solution phase structures. These structures may exhibit larger conformational fluctuations, leading to a net larger average distance between the ring and the amide proton compared to the X-ray structure as well as fluctuations in the direction of the magnetic dipole arising from the aromatic side chains and, therefore, a smaller ring current effect.

4. SUMMARY

We have presented sets of ring current intensity parameters for chemical shift predictions with the point-dipole, Haigh−Mallion, and Johnson−Bovey models. The maximum errors arising from use of the presented parameters are judged to be within ±0.1 ppm from what would have been computed at the CCSD(T)/CBS for a set of 932 test cases. Further improvements in computational methodology are thus not expected to yield any significant qualitative or quantitative improvement in chemical shift prediction in proteins. Preliminary calculations at the B3LYP/6-311++G(d,p) level using methane as a probe and intermolecular geometries corresponding to those in Table 3 suggest that the current parameters can be used to predict ring current effects on CH protons to within 0.2 ppm.

The presented parameters are rigorously based on the underlying physical properties of aromatic molecules. Parameters based on empirical models were found to perform worse on our amide proton test set. Our report of the superiority of a physics-based method over empirical methods is backed up by the fact that the parameters obtained by Case[18] through QM methods have the same disagreements with the empirical methods as our parameters, despite the fact that the computational methodology used by Case was somewhat different than ours.

Finally, we have made a detailed numerical comparison between the point-dipole, Haigh−Mallion, and Johnson−Bovey models. The chemical shift predictions of the three models were nearly identical, and no outliers compared to our quantum mechanical calculations were found in any of the three models. Apart from reported problems with predictions of ring current effects in macrocyclic rings, such as those found in porphyrins,[48] which should be a nonissue for most uses in protein chemical shift predictions, the three methods should yield results of identical accuracy. Hence, we suggest that the point-dipole model should be used in future chemical shift prediction software, since (1) it both is computationally faster than competing models, since it does not require any integral evaluation as opposed to the Johnson−Bovey model, and contains significantly fewer geometric terms than the Haigh−Mallion model and (2) it is much easier to implement than the competing models.

■ ASSOCIATED CONTENT

Supporting Information. Thorough descriptions of the point-dipole, Haigh−Mallion, and Johnson−Bovey models and our implementations are described. The material also includes sketches of the used molecules, linear correlation values and scatter plots of the fits used to obtain the intensity values of Table 4, and an additional investigation of the equilibrium geometry dependence of FMA NMR shielding calculations.

This information is available free of charge via the Internet at http://pubs.acs.org/.

AUTHOR INFORMATION

Corresponding Authors

*E-mail: andersx@nano.ku.dk; jhjensen@chem.ku.dk.

ACKNOWLEDGMENT

A.S.C. is recipient of a Ph. D. scholarship funded by the Novo Nordisk STAR program. S.P.A.S. thanks the Danish Center for Scientific Computing (DCSC), the Danish Natural Science Research Council/The Danish Council for Independent Research, and the Carlsberg Foundation for support.

REFERENCES

(1) Raman, S.; Lange, O.; Rossi, P.; Tyka, M.; Wang, X.; Aramini, J.; Liu, G.; Ramelot, T.; Eletsky, A.; Szyperski, T.; Kennedy, M. A.; Prestegard, J.; Montelione, G. T.; Baker, D. *Science* **2010**, *327*, 1014–1018.

(2) Meiler, J.; Baker, D. *Proc. Natl. Acad. Sci. U.S.A.* **2003**, *100*, 15404–15409.

(3) Jensen, M. R.; Salmon, L.; Nodet, G.; Blackledge, M. *J. Am. Chem. Soc.* **2010**, *132*, 1270–1272.

(4) Vila, J. A.; Scheraga, H. A. *Acc. Chem. Res.* **2009**, *42*, 1545–1553.

(5) Robustelli, P.; Kohlhoff, K. J.; Cavalli, A.; Vendruscolo, M. *Structure* **2010**, *18*, 923–933.

(6) Neal, S.; Nip, A. M.; Zhang, H.; Wishart, D. S. *J. Am. Chem. Soc.* **1991**, *111*, 9436–9444.

(7) Xu, X. P.; Case, D. A. *J. Biomol. NMR* **2001**, *21*, 321–333.

(8) Xu, X. P.; Case, D. A. *Biopolymers* **2002**, *65*, 408–423.

(9) Shen, Y.; Bax, A. *J. Biomol. NMR* **2007**, *38*, 289–302.

(10) Meiler, J. *J. Biomol. NMR* **2003**, *26*, 25–37.

(11) Haigh, C. W.; Mallion, R. B. *Prog. NMR Spectrosc.* **1980**, *13*, 303–344.

(12) Moon, S.; Case, D. A. *J. Biomol. NMR* **2007**, *38*, 139–150.

(13) Mulder, F. A.; Filatov, M. *Chem. Soc. Rev.* **2010**, *395*, 578–590.

(14) Monya, G.; Zauhar, R. J.; Williams, H. J.; Nachman, R. J.; Scott, A. I. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 702–709.

(15) Johnson, C. E.; Bovey, F. A. *J. Chem. Phys.* **1958**, *29*, 1012–1014.

(16) Pople, J. A. *J. Chem. Phys.* **1956**, *24*, 1111.

(17) Pople, J. A. *Mol. Phys.* **1958**, *1*, 175–180.

(18) Case, D. A. *J. Biomol. NMR* **1995**, *6*, 341–346.

(19) Boyd, J.; Skrynnikov, N. R. *J. Am. Chem. Soc.* **2002**, *124*, 1832–1833.

(20) Berman, J.; Westbrook, H. M.; Feng, Z.; Gilliland, G.; Bhat, T.; Weissig, H.; Shindyalov, I.; Bourne, P. *Nucleic Acids Res.* **2000**, *106*, 16972–16977.

(21) Dolinsky, T. J.; Czodrowski, P.; Li, H.; Nielsen, J. E.; Jensen, J. H.; Klebe, G.; Baker, N. A. *Nucleic Acids Res.* **2007**, *35*, 522–525.

(22) Dolinsky, T. J.; Nielsen, J.; McCammon, J. A.; Baker, N. A. *Nucleic Acids Res.* **2004**, *32*, 665–667.

(23) Bondi, A. *J. Phys. Chem.* **1964**, *68*, 441–451.

(24) Becke, A. D. *J. Phys. Chem.* **1993**, *98*, 5648–5652.

(25) Stephens, P.; Devlin, F.; Chabalowski, C.; Frisch, M. *J. Phys. Chem.* **1994**, *98*, 11623–11627.

(26) Moon, S.; Case, D. A. *Nucleic Acids Res.* **2004**, *32*, 665–667.

(27) Kupka, T.; Ruscic, B.; Botto, R. E. *J. Phys. Chem. A* **2002**, *106*, 10396–10407.

(28) Kupka, T.; Ruscic, B.; Botto, R. E. *Solid State Nucl. Magn. Reson.* **2003**, *23*, 145–167.

(29) Dunning, T. H. *J. Chem. Phys.* **1989**, *90*, 1007–1023.

(30) Marquardt, D. W. *J. Appl. Math.* **1963**, *11*, 431–441.

(31) Levenberg, K. *Q. Appl. Math.* **1944**, *2*, 164–168.

(32) Jensen, F. *J. Chem. Theory Comput.* **2008**, *5*, 719–727.

(33) Jameson, K. A.; Jameson, C. J. *Chem. Phys. Lett.* **1987**, *134*, 461–466.

(34) Vaara, J.; Kaski, J.; Joksaari, J.; Diehl, P. *J. Phys. Chem. A* **1997**, *101*, 5069–5081.

(35) Ruud, K.; Astrand, P.-O.; Taylor, P. R. *J. Am. Chem. Soc.* **2001**, *123*, 4826–4833.

(36) Sauer, S. P. A.; Spirko, V.; Paidarová, I.; Kraemer, W. P. *Chem. Phys.* **1997**, *214*, 91–102.

(37) Wigglesworth, R. D.; Raynes, W. T.; Sauer, S. P. A.; Oddershede, J. *Mol. Phys.* **1999**, *96*, 1595–1607.

(38) Wigglesworth, R. D.; Raynes, W. T.; Kirpekar, S.; Oddershede, J.; Sauer, S. P. A. *J. Chem. Phys.* **2000**, *112*, 736–746.

(39) Kern, C. W.; Matcha, R. L. *J. Phys. Chem.* **1968**, *49*, 2081–2092.

(40) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. A.; Cheeseman, J. R.; Montgomery, A. J., Jr.; Vreven, T.; Kudin, K. N.; Burant, J. C.; Millam, J. M.; Iyengar, S. S.; Tomasi, J.; Barone, V.; Mennucci, B.; Cossi, M.; Scalmani, G.; Rega, N.; Petersson, G. A.; Nakatsuji, H.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Klene, M.; Li, X.; Knox, J. E.; Hratchian, H. P.; Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.; Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Ayala, P. Y.; Morokuma, K.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.; Zakrzewski, V. G.; Dapprich, S.; Daniels, A. D.; Strain, M. C.; Farkas, O.; Malick, D. K.; Rabuck, A. D.; Raghavachari, K.; Foresman, J. B.; Ortiz, J. V.; Cui, Q.; Baboul, A. G.; Clifford, S.; Cioslowski, J.; Stefanov, B. B.; Liu, G.; Liashenko, A.; Piskorz, P.; Komaromi, I.; Martin, R. L.; Fox, D. J.; Keith, T.; Al-Laham, M. A.; Peng, C. Y.; Nanayakkara, A.; Challacombe, M.; Gill, P. M. W.; Johnson, B.; Chen, W.; Wong, M. W.; Gonzalez, C.; Pople, J. A. *Gaussian 03*; Gaussian, Inc.: Wallingford, CT, 2004.

(41) Ahlrichs, R.; Baer, M.; Haeser, M.; Horn, H.; Koelmel, C. *Chem. Phys. Lett.* **1989**, *162*, 165.

(42) Stanton, J. F.; Gauss, J.; Harding, M. E.; Szalay, P.; Auer, A. A.; Bartlett, R. J.; Benedikt, U.; Berger, C.; Bernholdt, D. E.; Bomble, Y. J.; Christiansen, O.; Heckert, M.; Heun, O.; Huber, C.; Jagau, T.-C.; Jonsson, D.; Juselius, J.; Klein, K.; Lauderdale, W. J.; Matthews, D. A.; Metzroth, T.; O'Neill, D. P.; Price, D. R.; Prochnow, E.; Ruud, K.; Schiffmann, F.; Stopkowicz, S.; Tajti, A.; Vazquez, J.; Wang, F.; Watts, J. D.; Almlöf, J.; Taylor, P. R.; Taylor, P. R.; Helgaker, T.; Jensen, H. J. A.; Jorgensen, P.; Olsen, J.; Mitin, A. V.; van Wüllen, C. CFOUR, a quantum chemical program package. For the current version, see http://www.cfour.de (accessed June 2010).

(43) London, F. *J. Phys. Radium* **1937**, *8*, 397–409.

(44) Ditchfield, R. *Mol. Phys.* **1974**, *27*, 789–807.

(45) Cheeseman, J. R.; Trucks, G. W.; Keith, T. A.; Frisch, M. J. *J. Chem. Phys.* **1996**, *104*, 5497–5509.

(46) Rablen, P. R.; Pearlman, S. A.; Finkbiner, J. *J. Phys. Chem. A* **1999**, *103*, 7357–7363.

(47) Ösapay, K.; Case, D. A. *J. Am. Chem. Soc.* **1991**, *111*, 9436–9444.

(48) Perkins, S. J. *Applications of ring current calculations to proton NMR of proteins and transfer RNA*; Plenum Press: New York, 1982; Vol. 4, Chapter 4, pp 193−336.

PHAISTOS: A Framework for Markov Chain Monte Carlo Simulation and Inference of Protein Structure

Wouter Boomsma, Jes Frellsen, Tim Harder, Sandro Bottaro, Kristoffer E. Johansson, Pengfei Tian, Kasper Stovgaard, Christian Andreetta, Simon Olsson, Jan B. Valentin, Lubomir D. Antonov, Anders S. Christensen, Mikael Borg, Jan H. Jensen, Kresten Lindorff-Larsen, Jesper Ferkinghoff-Borg, Thomas Hamelryck (2013) PHAISTOS: A framework for Markov chain Monte Carlo simulation and inference of protein structure. *Journal of Computational Chemistry*, 34:1697-1705.

PHAISTOS: A Framework for Markov Chain Monte Carlo Simulation and Inference of Protein Structure

Wouter Boomsma,[a,b]* Jes Frellsen,[a] Tim Harder,[a,c] Sandro Bottaro,[d,e] Kristoffer E. Johansson,[a] Pengfei Tian,[f] Kasper Stovgaard,[a] Christian Andreetta,[a,g] Simon Olsson,[a] Jan B. Valentin,[a] Lubomir D. Antonov,[a] Anders S. Christensen,[h] Mikael Borg,[a,i] Jan H. Jensen,[h] Kresten Lindorff-Larsen,[a] Jesper Ferkinghoff-Borg,[e] and Thomas Hamelryck[a]

We present a new software framework for Markov chain Monte Carlo sampling for simulation, prediction, and inference of protein structure. The software package contains implementations of recent advances in Monte Carlo methodology, such as efficient local updates and sampling from probabilistic models of local protein structure. These models form a probabilistic alternative to the widely used fragment and rotamer libraries. Combined with an easily extendible software architecture, this makes PHAISTOS well suited for Bayesian inference of protein structure from sequence and/or experimental data. Currently, two force-fields are available within the framework: PROFASI and OPLS-AA/L, the latter including the generalized Born surface area solvent model. A flexible command-line and configuration-file interface allows users quickly to set up simulations with the desired configuration. PHAISTOS is released under the GNU General Public License v3.0. Source code and documentation are freely available from http://phaistos.sourceforge.net. The software is implemented in C++ and has been tested on Linux and OSX platforms. © 2013 Wiley Periodicals, Inc.

DOI: 10.1002/jcc.23292

Introduction

Two methods dominate the field of molecular simulation: molecular dynamics (MD) and Markov chain Monte Carlo (MCMC). The main difference between the methods lies in the way the system is updated in each iteration. MD involves iterating between calculating the forces exerted on each particle in the system and using Newton's equations of motion to update their positions. In contrast, MCMC is a statistical approach, where the goal is to generate samples from a probability distribution associated with the system, typically a Boltzmann distribution. MD has generally been regarded as best-suited for exploring dense molecular systems such as the native ensemble of proteins, while MCMC methods can be more efficient for longer time scale simulations involving large structural rearrangements.[1] Using optimized move sets it has, however, been demonstrated that even in the densely packed native state, MCMC can serve as an efficient alternative to MD.[2–4] In addition, the statistical nature of MCMC methods make them particularly well-suited for Bayesian inference of protein structure from experimental data.[5]

The freedom in the choice of moves in Monte Carlo simulations means that there is potential progress to be made in designing new, improved move types, thereby further increasing the time scales and molecular sizes amenable to simulation. In this article, we present a software framework designed with this goal in mind. The PHAISTOS framework contains implementations of recently developed tools that increase the

[a] *W. Boomsma, J. Frellsen, T. Harder, KE. Johansson, K. Stovgaard, C. Andreetta, S. Olsson, JB. Valentin, L. D. Antonov, M. Borg, K. Lindorff-Larsen and T. Hamelryck*
Department of Biology, University of Copenhagen, Copenhagen, 2200, Denmark
E-mail: wb@bio.ku.dk

[b] *W. Boomsma*
Department of Astronomy and Theoretical Physics, University of Lund, Lund, SE-223 62, Sweden

[c] *T. Harder*
Center for Bioinformatics, University of Hamburg, Hamburg, 20146, Germany

[d] *S. Bottaro*
Scuola Internazionale Superiore di Studi Avanzati, Trieste, 34136, Italy

[e] *S. Bottaro and J. Ferkinghoff-Borg*
Department of Biomedical Engineering, DTU Elektro, DTU, Kongens Lyngby, 2800, Denmark

[f] *P. Tian*
Niels Bohr Institute, University of Copenhagen, Copenhagen, 2100, Denmark

[g] *C. Andreetta*
Computational Biology Unit, Uni Computing, Uni Research, Norway

[h] *AS. Christensen and JH. Jensen*
Department of Chemistry, University of Copenhagen, Copenhagen, 2100, Denmark

[i] *M. Borg*
BILS, Science for Life Laboratory, Box 1031, Solna, 171 21, Sweden

Contract/grant sponsor: Danish Council for Independent Research; Contract/grant numbers: FNU272-08-0315 (to W.B.); FTP274-06-0380 (to K.S.); FTP09–066546 (to S.O. and J.V.), and FTP274-08-0124 (to K.E.J.).
Contract/grant sponsor: Danish Council for Strategic Research; Contract/grant number: NABIIT2106-06-0009.
Contract/grant sponsors: Novo Nordisk STAR Program (to A.S.C.), Novo Nordisk Foundation (to K.L.L.), and Radiometer (DTU) (to S.B.).

efficiency and scope of MCMC-based simulations. Through a modular design, the software can easily be extended with new move types and force-fields, making it possible to experiment with novel Monte Carlo strategies. Finally, using flexible configuration file and command line options, users can quickly set up simulations with any combination of moves, energy terms, and other simulation settings.

By making our methods available in an easily extendible, open source framework, we hope to further encourage the use of MCMC for protein simulations and promote the development of new MCMC methodologies for the simulation, prediction, and inference of protein structure.

Methodology

The framework is split into four main types of components: moves, energy terms, observables, and Monte Carlo methods. For each of these types, a number of algorithms are available. Moves and energies are normally used in sets: a weighted set of moves is referred to as a move collection, while an energy function is composed of a weighted sum of energy terms. Observables are similar to energy terms, but are typically only evaluated at certain intervals to extract statistics during a simulation. In the following description, each algorithm is annotated with its corresponding command line option name in a monospace font.

Moves

One of the main distinguishing features of the PHAISTOS package is efficient sampling, obtained through an elaborate set of both established and novel Monte Carlo moves. Each move stochastically modifies a protein chain in a specific way. Weighted sets of these moves can be selected from the command line, allowing the user to easily experiment and fine-tune the set of moves for a given simulation scenario. All moves in PHAISTOS can be applied such that detailed balance is obeyed, which ensures, if the sampling is ergodic, that simulations sample from a well-defined target distribution (e.g., the canonical or multicanonical ensemble).

The framework contains many of the established moves from the literature, including various pivot moves (`move-pivot-uniform`, `move-pivot-local`), the crankshaft/backrub local move (`move-crankshaft`),[6,7] the CRA local move (`move-cra`),[8] and the semilocal biased Gaussian step (BGS) (`move-semilocal`).[9] Side-chain conformational sampling can be done either from Gaussian distributions given by rotamer libraries (`move-sidechain-rotamer`)[10] or through Gaussians centered around the current side-chain conformation (`move-sidechain-local`).

Moves using Probabilistic Models. PHAISTOS has broad support for sampling using biased proposals. Usually, if an MCMC simulation were to be conducted without the presence of a force-field, a uniform distribution in configurational space would be obtained. In the case of biased sampling, moves are instead allowed to follow a specific distribution during the simulation. This bias can then, optionally, be divided out, so that it does not influence the final statistical ensemble, but only serves to increase sampling efficiency by focusing on the most important regions of conformational space. If the bias is left in, it corresponds to an implicit extra term in the energy function.

Typically, the bias is chosen to reflect prior knowledge about the local structure of the molecule. A good example is the common use of fragment and rotamer libraries for structure prediction.[11,12] These methods are used strictly for sampling, and the introduced bias is not easily quantifiable, which also makes it difficult to ensure detailed balance for the Markov chain. In contrast, PHAISTOS includes a number of moves based on probabilistic models, which support both sampling of conformations and the evaluation of the bias introduced with those moves. This makes them uniquely suited for use in MCMC simulations.

Four different structural, probabilistic models are available: FB5HMM models the Cα trace of a protein,[13] COMPAS models a reduced single-particle representation of amino acid side-chains, whereas TORUSDBN and BASILISK, respectively, model backbone and side-chain structure in atomic detail.[14,15] All models can be applied both as proposal distributions in the form of Monte Carlo moves (`move-backbone-dbn`, `move-sidechain-basilisk`, `move-sidechain-compas`) and as probabilistic components of an energy function (`energy-backbone-dbn`, `energy-basilisk`, `energy-compas`). Figure 1 illustrates how dihedral angles are sampled from a TORUSDBN-like model of the protein backbone. The practical details on how probabilistic models can be incorporated in an energy function are discussed in the "Energies" section.

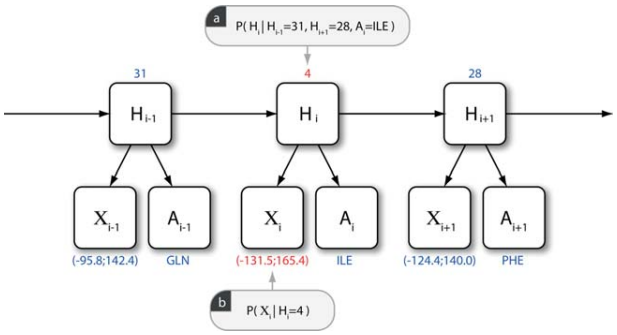


Figure 1. An illustration of a simplified version of the TORUSDBN model of backbone local structure, showing the architecture of the dynamic Bayesian network (DBN) and an example of values for the individual nodes. Each *A* node is a discrete distribution over amino acids, whereas each *X* node is a bivariate distribution over (ϕ, ψ) angle pairs. The hidden node (*H*) sequence is a Markov chain of discrete states, representing the sequence of residues in a protein chain. Each hidden node state corresponds to a particular distribution over angle pairs and amino acid labels. The values highlighted in red are the result of a single resampling step of the (ϕ, ψ) angle pair at some position *i* in the chain: a) The hidden node state H_i is resampled based on the current values of the values of neighboring *H* values and the amino acid label at position *i* ($P(H_i|H_{i-1}, H_{i+1}, A_i) \propto P(H_i|H_{i-1})P(H_{i+1}|H_i)P(A|H_i)$); b) A (ϕ, ψ) value is drawn from the bivariate angular distribution corresponding to the sampled *H* value ($P(X_i|H_i)$). A full description of the TORUSDBN model can be found in the original publication.[14]

Efficient Local Updates. An important challenge in Monte Carlo simulations is to ensure efficient sampling in dense states where proposed conformational changes will have a high probability of containing self-collisions. In particular, pivot moves will typically have very poor acceptance rates in this scenario. The solution is typically to expand the move set to include local moves, which only change the atom positions within a small segment of the chain.

In addition to various established local move methods from the literature, PHAISTOS includes a novel method, called CRISP[4] (`move-crisp`), which is particularly well-suited for this problem. Unlike other local move approaches,[7,8] CRISP is able to generate updates to a segment of the chain without disrupting its local geometry. Often, local move algorithms are designed as a two-step process, where some angular degrees of freedom are modified stochastically (prerotation), whereas others are modified deterministically to bring the chain back to a closed state (postrotation). From the work of Gō and Scheraga,[16] it is known that in general, six degrees of freedom are required for the postrotation step. CRISP distinguishes itself from previous methods by merging these two steps, modifying the stochastic prerotation step so that it takes the resulting postrotation step into account. More precisely, for each application of the move, a random segment of the protein is selected, and a multivariate Gaussian distribution is constructed over the angular change in the $n-6$ prerotation degrees of freedom $\delta\bar{\chi}_{\text{pre}}$

$$P(\delta\bar{\chi}_{\text{pre}}) \propto \exp\left(-\frac{1}{2}\delta\bar{\chi}_{\text{pre}}^{T}\lambda\left(\mathbf{C}_{n-6}+\mathbf{S}^{T}\mathbf{C}_{6}\mathbf{S}\right)\delta\bar{\chi}_{\text{pre}}\right). \quad (1)$$

Here, **C** is an inverse diagonal covariance matrix specifying the desired fluctuations for the individual angular degrees of freedom, **S** is a linear transformation mapping the prerotational degrees of freedom to the corresponding postrotational values, and λ is a scaling parameter determining the size of the move. In effect, to first order, the method samples from a distribution of closed chain structures, ensuring high-quality local structure in all samples. We have recently shown that this has a dramatic impact on simulation performance, in particular for dense molecular systems.[4]

Low acceptance rates are also sometimes observed in side-chain moves. When using a fine-grained force-field such as OPLS-AA/L, we have experienced that the standard resampling of side-chains can be overly intrusive. Particularly in the case of side-chains involved in several simultaneous hydrogen bonds, traditional moves tend to break all hydrogen bonds at once, typically leading to the rejection of such updates. To avoid this problem, PHAISTOS includes a novel move (`side-chain-local`) that, for a given side-chain, randomly selects an atom that potentially participates in hydrogen bonds, and constrains its position using a technique similar to that of the semilocal BGS backbone move.[4,9]

Energies

Two established force-fields are currently implemented within the framework: the PROFASI force-field[17] and the OPLS-AA/L[18] force-field in combination with the generalized Born surface area (GB/SA) implicit solvent model.[19] These represent two extremes in the range of force-fields available in the literature: an ultrafast force-field modeling effective interactions in the presence of a solvent and a classic fine-grained molecular mechanics force-field combined with a more accurate implicit solvent model. The two force-fields were selected to provide support for a broad range of simulation tasks. The efficiency of the PROFASI force-field makes it possible readily to conduct reversible folding simulations of peptides and small proteins.[17] The OPLS-AA/L force-field in combination with the GB/SA solvent model is more accurate, but also significantly slower, and is typically used for exploring the details of native ensembles. It can also be used for structure refinement, for instance of structures obtained in a reversible folding simulation using PROFASI. For increased efficiency, all nonbonded force-field terms in both force-fields have been implemented using the chaintree data structure,[20] which avoids recalculation of energy contributions that are not modified in a given iteration of the simulation. Together with effective local moves, this can result in a considerable computational speed-up.

PROFASI. The PROFASI force-field consists of four terms[17]

$$E=E_{\text{ev}}+E_{\text{hb}}+E_{\text{sc}}+E_{\text{loc}} \quad (2)$$

where E_{ev} captures excluded volume effects, E_{hb} is a hydrogen bond term, E_{sc} is a side-chain interaction term, and E_{loc} concerns the local interactions along the chain. The excluded volume potential is a simple r^{-12} interaction between all atom pairs, where r denotes the distance between the atoms. The strength of a hydrogen bond in PROFASI depends on the detailed geometry of the bond, parameterized through the N—H—O and H—O—C angles. The side-chain potential consists of a charge–charge and a hydrophobicity contribution. For each residue pair, these consist of a product between a conformation-dependent contact strength and an energy that depends on the specific amino acid types involved in the bond. Finally, the local energy term captures interactions between partial charges in neighboring peptide units along the chain, with a correction term for improved consistency with the Ramachandran plot, and a side-chain torsion potential. As bond angles and bond lengths are assumed fixed during PROFASI simulations, no further local interactions are included.

A distinguishing feature of the PROFASI force-field is the presence of a global interaction cutoff of 4.5 Å. Although this necessarily excludes various long-range interactions, it is also one of the main reasons behind the efficiency of the force-field. Despite this restriction, the force-field has been demonstrated to successfully fold a range of peptides and small proteins,[17] while still being fast enough for many-body aggregation simulations.[21,22]

OPLS-AA/L. In contrast to PROFASI, the OPLS-AA/L force-field includes local terms for bond angles, bond lengths, and torsions. The bond angle and bond length potentials are simple harmonic terms, whereas the torsion term has the form[18]

$$E_{\text{torsion}} = \sum_i \sum_{j=1}^{3} w_j \left(1 + (-1)^{j+1} \cos(j\theta_i)\right) \quad (3)$$

where the outer sum iterates over all dihedrals θ_i. The nonbonded interactions include standard Lennard-Jones and Coulomb potentials

$$E_{\text{nb}} = \sum_{i>j} w_{ij} \left(4\epsilon_{ij} \left(\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right) + \frac{1}{4\pi\varepsilon_0}\frac{q_i q_j}{r_{ij}}\right) \quad (4)$$

where r_{ij} is the distance between atoms i and j, q_i and q_j are the corresponding partial charges, ε_0 is the vacuum permittivity, and σ_{ij} and ϵ_{ij} are calculated using the combination rules $\sigma_{ij}=\sqrt{\sigma_i\sigma_j}$ and $\epsilon_{ij}=\sqrt{\epsilon_i\epsilon_j}$, respectively. Finally, w_{ij} works to exclude interactions between atoms that are separated by only a few covalent bonds. Thus, $w_{ij}=0.0$ for direct neighbors (1,2) and pairs separated by a single other atom (1,3), $w_{ij}=0.5$ for pairs separated by two atoms (1,4), and $w_{ij}=1.0$ for all others.

Our implementation of OPLS-AA/L follows that of the Tinker simulation package.[23] We ensured that energies produced by our program match those obtained when running Tinker.

GB/SA. The PROFASI force-field is parameterized to capture effective interactions in the presence of a solvent. In contrast, OPLS-AA/L should be combined with a suitable solvent model to reproduce physiological conditions. To model the effect of the solvent on hydrophobic interactions and electrostatics, we use the OPLS-AA/L force-field in combination with the GB/SA implicit solvent model.[19]

Many implicit solvent models express the solvation free energy G_{solv} as a sum of nonpolar and electrostatic contributions

$$G_{\text{solv}} = G_{\text{npol}} + G_{\text{pol}} \quad (5)$$

Here, G_{npol} is the free energy of solvating the molecule with all the partial charges set to zero, and G_{pol} is the reversible work required to increase the charges from zero to their full values.[24] In GB/SA, the nonpolar contribution G_{npol} is assumed to be proportional to the solvent accessible surface area, while the generalized Born approximation is used to calculate the electrostatic solvation energy using the pairwise summation[25]

$$G_{\text{pol}} = -\frac{1}{8\pi\varepsilon_0}\left(1 - \frac{1}{\varepsilon}\right)\sum_{i,j}^{n} \frac{q_i q_j}{f_{\text{GB}}} \quad (6)$$

where ε is the dielectric constant of the solvent, and q_i is the partial charge of atom i. $f_{\text{GB}} = \sqrt{r_{ij}^2 + \alpha_i\alpha_j \exp\left(-r_{ij}^2/4\alpha_i\alpha_j\right)}$ is a function of the distance r_{ij} and of the so-called Born radii α, which reflects the average distance of the charged atom to the dielectric medium. For our implementation, the Born radii are calculated using an analytical expression proposed by Still and coworkers.[19]

Incorporating Probabilistic Models in the Energy Function. When using moves that are based on probabilistic models such as TORUSDBN and BASILISK, it gives rise to a bias in the simulation, which can be regarded as an implicit energy term. In PHAISTOS, the energy contributions of these probabilistic models can also be evaluated explicitly, by adding them as a term to the energy function. This makes it possible to use the probabilistic models as energies in a simulation with a standard set of unbiased moves, or to compensate for the bias of a move by adding the corresponding energy term with negative weight. When used as energies, the values are reported in minus log-probabilities. To facilitate the combination of classic energy terms with probabilistic terms, the energies of physical force-fields such as PROFASI and OPLS-AA/L are likewise reported as minus log-probabilities: they are multiplied by $-1/kT$, where T is the simulation temperature and k is the Boltzmann constant.

As an example, for the TORUSDBN-like model in Figure 1, the log-likelihood for a given state is

$$\begin{aligned} LL(\bar{X},\bar{A}) &= \ln \sum_{\bar{H}} P(\bar{X},\bar{A},\bar{H}) \\ &= \ln \sum_{\bar{H}} P(X_1|H_1)P(A_1|H_1)P(H_1)\prod_{i=2}^{N} P(X_i|H_i)P(A_i|H_i)P(H_i|H_{i-1}) \end{aligned} \quad (7)$$

where $\bar{X}$, $\bar{A}$, and $\bar{H}$ are the sequences of angle pairs, amino acid labels, and hidden node labels, respectively, i is the residue index, and N is the sequence length. Each hidden node label is the index of a component of the emission distributions of the model. For instance, the Ramachandran distribution is modeled as a weighted sum of bivariate von Mises distribution components.[14] The hidden nodes are "nuisance" parameters and are therefore summed out in the evaluation of the likelihood. Note that the sum runs over all possible hidden node *sequences*, a calculation that can be done efficiently using dynamic programming.[14]

Observables

Observables in PHAISTOS allow a user to extract information about the current state of a simulation. Examples of observables include root-mean-square-deviation (RMSD) (`observable-rmsd`) and radius of gyration (`observable-rg`). In addition, all energy terms are also available as observables. A user can specify a selection of observables from the command line or settings file, choosing how frequently they should be registered and in which format. While most observables will return a single value, others have more elaborate outputs, such as dumping of complete structural states to PDB files (`observable-pdb`) or to a molecular trajectory file in the Gromacs XTC format (`observable-xtc-trajectory`).[26] Finally, observables can be dumped to the header or as *b*-factors in outputted PDB-files. The latter makes it possible to annotate structures with residue-specific information, such as the number of contacts, degree of burial, and more sophisticated evaluations of the environment of each residue.[27]

Monte Carlo

Although PHAISTOS can be used for Monte Carlo minimization, the primary focus of the framework is MCMC simulation,

where the goal is to produce samples from the Boltzmann distribution corresponding to a given force-field at a specified temperature. All moves in PHAISTOS are, therefore, designed to be compatible with the property of detailed balance, in the sense that their proposal probabilities can be evaluated. That is, for a move from state x to state x'

$$\pi(x)P(x \to x') = \pi(x')P(x' \to x) \tag{8}$$

where $\pi(x)$ is the stationary distribution, and $P(x \to x')$ is the probability of moving from state x to x' using a given move. Factoring $P(x \to x')$ into a "selection" probability P_s and an "acceptance" probability P_α, we have

$$\frac{P_\alpha(x \to x')}{P_\alpha(x' \to x)} = \frac{\pi(x')P_s(x' \to x)}{\pi(x)P_s(x \to x')} \tag{9}$$

Most of the moves are symmetric, in the sense that $P_s(x' \to x)/P_s(x \to x') = 1$. However, for moves such as the local and semilocal moves, this is not the case, and it is important that this bias be correctly compensated for. Implementation-wise, each Move object is responsible for calculating the bias that it introduces, and the Monte Carlo class will then compensate for it when necessary.

Metropolis-Hastings. The most common way to ensure that eq. (9) is fulfilled is to use the Metropolis-Hastings (MH) acceptance criterion

$$P_\alpha(x \to x') = \min\left(1, \frac{\pi(x')P_s(x' \to x)}{\pi(x)P_s(x \to x')}\right) \tag{10}$$

This is the default simulation method used in PHAISTOS (`monte-carlo-metropolis-hastings`). It is useful for exploring near-native ensembles and can be efficient when simulating at the critical temperature of a system. However, for more complicated systems, MH simulations tend to spend excessive periods of time exploring local minima, leading to poor mixing, and, therefore, slow convergence.

Generalized Ensembles. To avoid the mixing problems associated with standard MH simulations, PHAISTOS includes support for conducting simulations in generalized ensembles.[28] Rather than sampling directly from the Boltzmann distribution, the central idea is to generate samples from a modified distribution, and subsequently reweight the obtained statistics to the Boltzmann distribution at a desired temperature. The acceptance criterion becomes

$$P_\alpha(x \to x') = \min\left(1, \frac{w(x')P_s(x' \to x)}{w(x)P_s(x \to x')}\right) \tag{11}$$

for a given weight function $w(x)$. The typical choice is the multicanonical ensemble,[29] corresponding to a flat distribution over energies. That is, $w(x) = 1/g(E(x))$, where $E(x)$ is the energy associated with conformational state x, and g is the number of states associated with a given energy (density of states). Another example is the $1/k$ ensemble, which attempts to provide ergodic sampling while maintaining primary focus on the low energy states.[30] In this case, the weight function is $w(x) = 1/k(E(x))$, where $k(E(x)) = \int_{-\infty}^{E(x)} g(\hat{E})d\hat{E}$. It can be shown that this is approximately equivalent to a flat histogram over $\ln(E(x))$.[30]

We have recently developed an automated method, MUNINN, for estimating the weights w in generalized ensemble simulations (http://muninn.sourceforge.net/). It employs the generalized multihistogram equations,[31] and uses a non-uniform adaptive binning of the energy space, ensuring efficient scaling to large systems. In addition, MUNINN allows weights to be restricted to cover a limited temperature range of interest. The MUNINN functionality is seamlessly integrated into PHAISTOS and can be activated by selecting the corresponding Monte Carlo engine (`monte-carlo-muninn`).

Monte Carlo Minimization. PHAISTOS contains a few simulation algorithms that are directed at optimization, rather than sampling. These include a simulated annealing class (`monte-carlo-simulated-annealing`) and a greedy Monte Carlo optimization class (`monte-carlo-greedy-optimization`), which are useful in cases where the user is interested in a single low-energy structure, rather than a full structural ensemble.

Program design

The framework is designed to be modular, both in software design and in the choices exposed to the user from the command line or settings file. As illustrated in the UML diagram in Figure 2, all energy terms are derived from the same base class and implement the same interface. Energy functions can easily be constructed from the command line or settings file by including the energy terms of interest. Moves and Monte Carlo simulation algorithms are structured in a similar way. This design makes it straightforward to implement new energy terms, moves or simulation algorithms with little knowledge of the overall code. Iterators are provided for easy iteration over atoms or residues in a molecule. In addition, caching and rapid determination of interacting atom pairs is made possible by an implementation of the chaintree algorithm.[20]

Finally, through a modular build-system, developers can readily write their own modules utilizing the library. Modules are separate code entities that are autodetected by the build system when present and can be enabled and disabled at compile time, making it easy to share code among collaborators.

Example

We include a step-by-step walk-through of the PHAISTOS simulation process. The goal is to conduct a reversible folding simulation of the 20-residue beta3s peptide,[32] demonstrating several of the features described earlier.

The user interface of PHAISTOS is designed to make it as easy as possible to set up simulations. Almost all options have default values, and it is, therefore, usually sufficient to supply only a few input options to the program. The program behavior can then gradually be fine-tuned using additional options in the configuration file later on. For this particular example, we use the following command from the command line

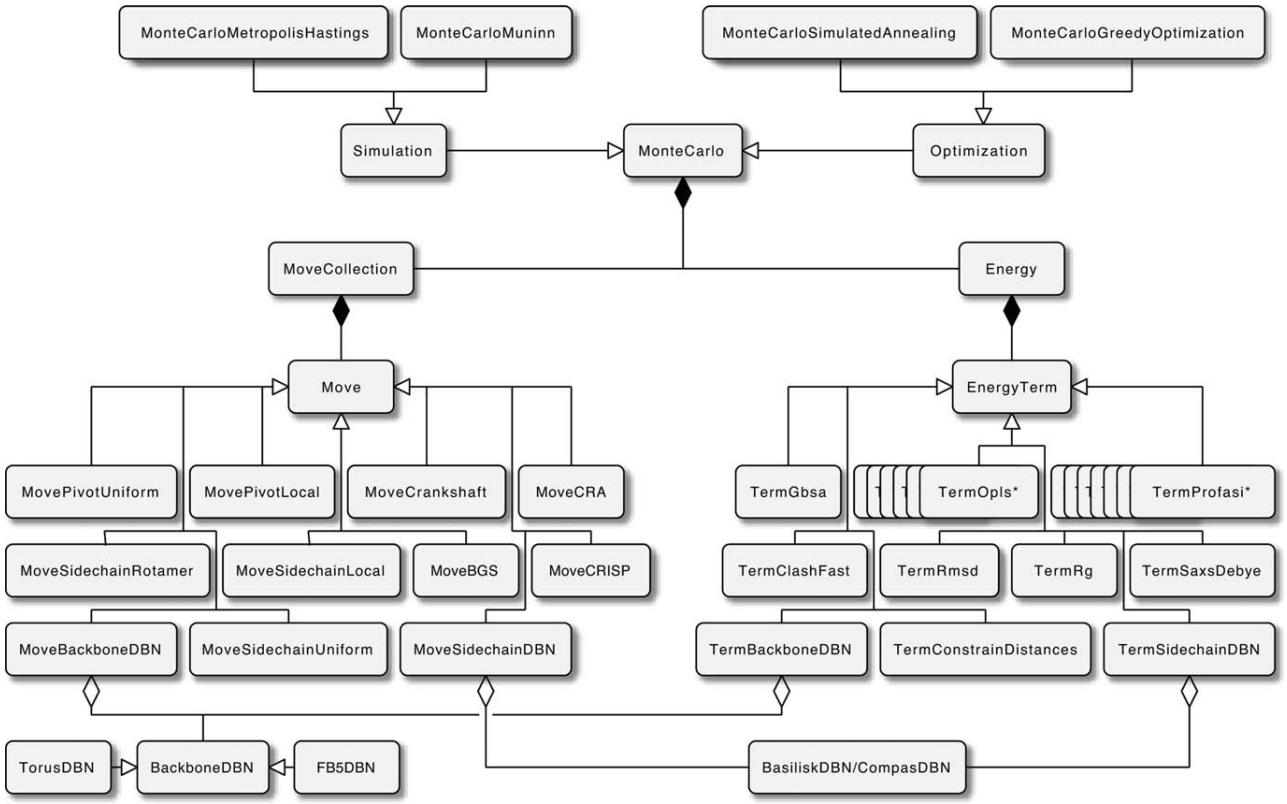


Figure 2. A UML-diagram of the major classes in the PHAISTOS library (black diamond: composition, white diamond: aggregation, arrow: inheritance). A Monte Carlo simulation object contains a MoveCollection object, which consists of a selection of moves, and an Energy object, which comprises a number of energy terms (TermOpls* and TermProfasi* denote the entire set of OPLS and PROFASI energy terms, respectively). Note that the probabilistic models (BackboneDBN/BasiliskDBN/CompasDBN) are available both as energy terms and as moves. A detailed description of all classes can be found in the Doxygen documentation on the PHAISTOS web site.

```
$ ./phaistos --aa-file beta3s.aa \
             --energy profasi-cached backbone-dbn[weight:-1] \
             --move backbone-dbn sidechain-uniform semilocal-dbn-eh \
             --threads 8 --temperature 283 \
             --monte-carlo muninn[min-beta:0.6,max-beta:1.1] \
             --observable backbone-dbn rmsd[reference-pdb-file:beta3s.pdb] \
             --observable xtc-trajectory
```

The aa-file argument specifies that we are reading an amino acid sequence from a file. The energy and move options select the relevant energy terms and moves, respectively. In this case, we use a cached version of the PROFASI force-field, and sample using TORUSDBN moves, uniformly distributed side-chain moves, and BGS moves using the TORUSDBN as a prior. We specify that the simulation should be conducted in eight parallel threads and set the temperature to 283 K. MUNINN is chosen to be the Monte Carlo engine, using a β (inverse temperature) range of $[0.6; 1.1]$. These β factors are unit-less, specified relative to the inverse temperature, and thus correspond to a temperature range of $\left[(1.1 \cdot 1/283\text{K})^{-1}; (0.6 \cdot 1/283\text{K})^{-1}\right] = [257\text{K}; 472\text{K}]$. Finally, we specify that we wish to record observables about the backbone-dbn energy, the RMSD to the native state, and dump structures to an XTC trajectory file. Apart from the RMSD observable, we do not provide the program with any information about the structure of the protein, and the simulation will, therefore, start in a random extended state.

To illustrate the framework's support for various types of biased sampling, this example uses a variant where the TORUSDBN bias is included in the sampling, and explicitly subtracted in the energy (i.e., the weight: -1 option of backbone-dbn). This means that the bias cancels out when extracting statistics at $\beta=1$, thus producing unbiased estimates at 283K. The simulation will produce a flat histogram over the expected energy range corresponding to the specified temperature range ($[\langle E\rangle_{257\text{K}}; \langle E\rangle_{472\text{K}}]$).

We ran PHAISTOS with the settings above on an eight-core 3.4 GHz Intel Xeon processor for 1 week. Figure 3 gives an overview of the results. The free energy plot in Figure 3a shows the distribution of energy versus RMSD extracted directly from the samples dumped during the simulation. As the samples were generated using a generalized ensemble

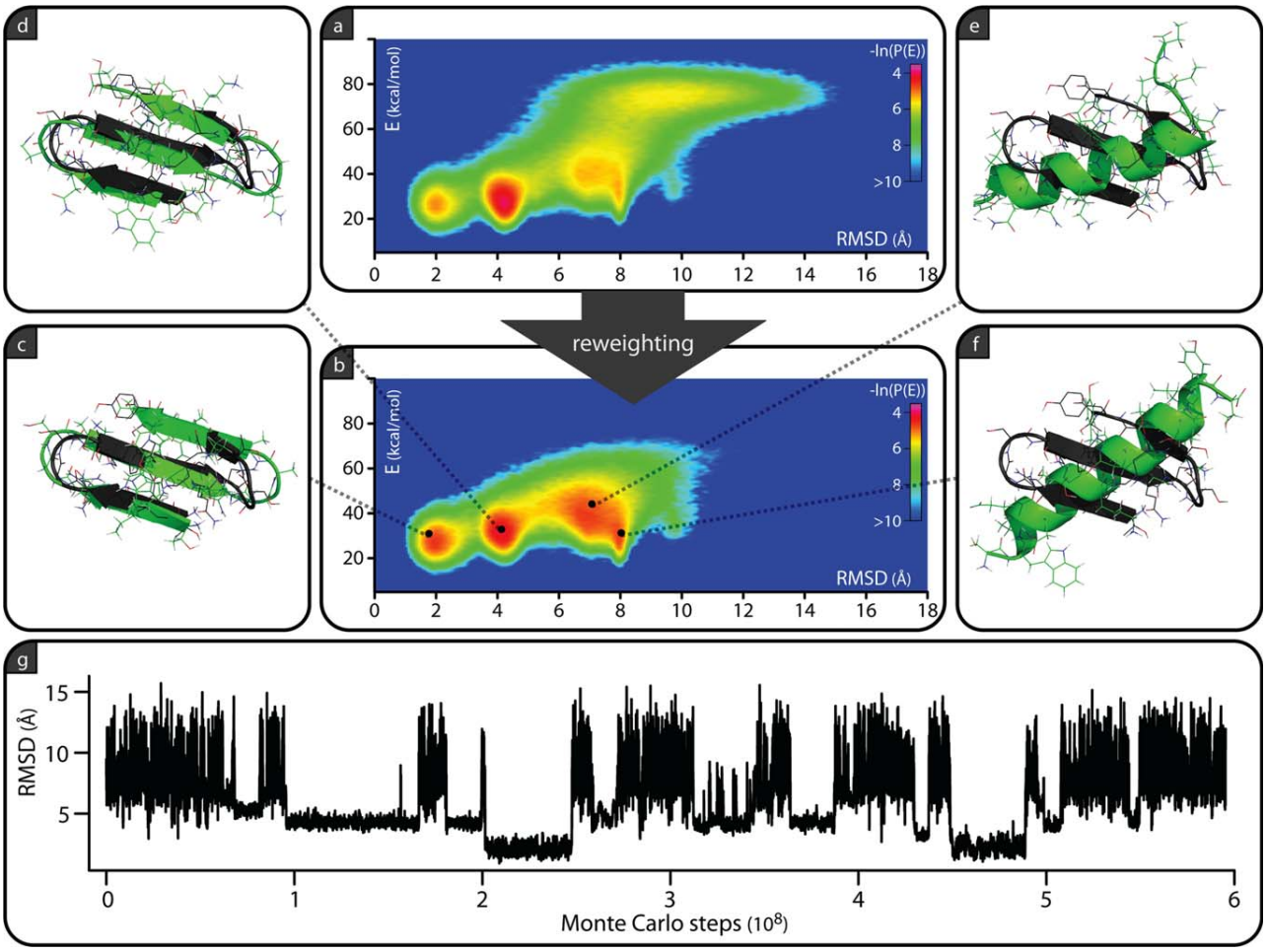


Figure 3. Illustration of a reversible folding simulation of the beta3s peptide in PHAISTOS. The simulation was conducted with the PROFASI force-field, using the MUNINN multihistogram method and a set of moves including TORUSDBN as a dihedral proposal distribution. The bias introduced by TORUSDBN is compensated for to ensure correctly distributed samples. a) Free energy plot as a function of energy and RMSD in the multicanonical (flat histogram) ensemble. b) Free energy plot as a function of energy and RMSD, reweighted to the canonical ensemble at 283 K. c–f) Representative cluster medoids found with reweighted clustering using the PLEIADES module, compared to the native structure[32] (shown in black). Figures created using Pymol.[33] g) RMSD versus time of one of the eight threads in the simulation.

technique, they must be reweighted to retrieve the statistics according to the Boltzmann distribution at the specified temperature. This is done using a script included in the MUNINN module, resulting in the plot in Figure 3b.

To find representative structures in the ensemble, we use the PLEIADES clustering module,[34] included in the framework. Again, it is important to remember that the raw data are produced in a generalize ensemble setting and must be reweighted. We ran the RMSD-based weighted *k*-means method implemented in the PLEIADES module to select the highlighted structures in Figure 3.

From the analysis above, we conclude that at 283 K, the protein is marginally stable in the PROFASI force-field, with native-like populations at 2 and 4 Å RMSD, but also a significant population of unstructured or helical conformations. These results are in approximate agreement with experiments, which suggest a folded population of between 13 and 31%.[32] This result is compatible with a previously published simulation of the same protein using an unbiased simulation technique.[17]

Results

To illustrate the versatility of PHAISTOS, we highlight several recently published applications of the framework.

Structure prediction and inference

An example of the applicability of PHAISTOS in the context of protein structure prediction is found in a recent study on potentials of mean force.[35] The study demonstrates how probabilistic models of local protein structure such as TORUSDBN and BASILISK can be combined with probabilistic models of nonlocal features, such as hydrogen bonding and compactness, using a simple probabilistic technique.

The framework has also been applied for inference of protein structure from small-angle X-ray scattering (SAXS) and nuclear magnetic resonance (NMR) experimental data. SAXS data contain low-resolution information on the overall shape of a protein, which can be useful for determining the relative domain positions and orientations in multidomain proteins or complexes.

This can for instance be used to infer structural models of multi-domain proteins connected by flexible linkers, given the atomic structures of the individual domains. Such calculations require efficient back-calculation of SAXS curves, which is made possible through a coarse-grained Debye method.[36,37]

NMR experimental data can provide high-resolution structural information that can improve the accuracy of a simulation. PHAISTOS contains preliminary support for sampling conditional on chemical shift data, which is known to contain substantial information on the local structure of a protein.[38,39] Furthermore, the framework was recently used for inferential structure determination using pair-wise distances obtained from NOE experiments, with TORUSDBN and BASILISK as prior distribution for the protein's backbone and side chains.[40]

Efficient clustering

Efficiently clustering a large number of protein structures is an important task in protein structure prediction and analysis. Typically, clustering programs require costly RMSD calculations for many pairs in the set of structures. PHAISTOS contains a clustering module called PLEIADES that uses a *k*-means clustering approach[41] to reduce the number of pair-wise RMSD distance calculations. Furthermore, PLEIADES includes support for replacing the RMSD distance computations with distances between vectors of Gauss integrals,[42] which provides dramatic computational speedups.[34]

Native ensembles

The energy landscape around the native state tends to be rugged, making it challenging to sample such states efficiently.[1] For these tasks, the CRISP backbone move is particularly well suited, given its ability to propose subtle, nondisruptive updates to the protein backbone. Monte Carlo simulations using this move were recently shown to explore conformational space with an efficiency on par with MD, outperforming the current state-of-the-art in local Monte Carlo move methods.[4]

The TYPHON module[43] rapidly explores near-native ensembles using the CRISP move in combination with a user-defined set of nonlocal restraints. Local structure is under the control of probabilistic models of the backbone (TORUSDBN) and side chains (BASILISK), while nonlocal interactions such as hydrogen bonds and disulfide bridges are heuristically imposed as Gaussian restraints. TYPHON can be seen as a "null model" of conformational fluctuations in proteins: it rapidly explores the conformational space accessible to a protein given a set of specified restraints.

Discussion

The relevance of a new software package should be assessed relative to already existing packages in the literature. We acknowledge that in our case, there are a number of such alternatives already available. We describe the most important ones here, focusing on the differences to the framework presented in this article.

Of the available Monte Carlo software packages, the ROSETTA package[11] is perhaps the most widely used and has an impressive track record for protein structure prediction and design.[44] The package focuses primarily on structure/sequence prediction (optimization) rather than simulation, and consequently, many of the moves in ROSETTA are not compatible with the property of detailed balance.

PHAISTOS also has some overlap with the PROFASI simulation package,[45] in the sense that both implement the BGS move[9] and the PROFASI energy function.[17] The PROFASI simulation program was designed as a tool for studying protein aggregation and is thus highly optimized for many-chain simulation using their lightweight force-field and under the assumption of fixed bond angles. PHAISTOS aims to provide a greater flexibility in the choice of energies and a wider selection of moves and is not limited to a fixed bond-angle representation.

The closest alternatives to PHAISTOS are perhaps the CAMPARI software package[1] and the Monte Carlo package in CHARMM,[2] which both provide functionality for conducting MCMC simulations using various force-fields and moves. Compared with PHAISTOS, the selection of force-fields and moves differ, and the focus is different. For instance, PHAISTOS has a strong focus on sampling using probabilistic models of local structure, which is not supported by either of the two alternatives.

The current version of the PHAISTOS framework has several limitations. To a user familiar with MD software, the primary limitation will presumably be the lack of explicit solvent models in the framework. The large conformational moves that provide the sampling advantage of Monte Carlo simulations are difficult to combine with an explicit solvent representation. In line with other Monte Carlo simulation packages, PHAISTOS is, therefore, currently limited to implicit solvent simulations. Another limitation is that PHAISTOS can currently only simulate a single polypeptide at a time. This restriction will be removed in the next release of the software, which will also include implementations of several new force-fields.

As the list of applications demonstrates, even in its current form, the framework provides the necessary tools for conducting relevant MCMC simulations of protein systems. The framework incorporates generalized ensembles and novel Monte Carlo moves, including moves that incorporate structural priors as proposal distributions. These features are unique to this framework and have been shown to increase sampling efficiency considerably.

The software is freely available under the GNU General Public License v3.0. All source code is fully documented using the Doxygen system (http://www.doxygen.org), and a user manual is available for detailed descriptions on how to set up simulations. Both sources of information are accessible via the PHAISTOS web site, http://phaistos.sourceforge.net.

Keywords: Markov chain Monte Carlo simulation · protein structure · probabilistic models · local moves · conformational sampling

How to cite this article: W. Boomsma, J. Frellsen, T. Harder, S. Bottaro, K. E. Johansson, P. Tian, K. Stovgaard, C. Andreetta, S. Olsson, J. B. Valentin, L. D. Antonov, A. S. Christensen, M. Borg, J. H. Jensen, K. Lindorff-Larsen, J. Ferkinghoff-Borg, T. Hamelryck, *J. Comput. Chem.* **2013**, *34*, 1697–1705. DOI: 10.1002/jcc.23292

[1] A. Vitalis, R. V. Pappu, *Annu. Rep. Comput. Chem.* **2009**, *5*, 49.
[2] J. Hu, A. Ma, A. R. Dinner, *J. Comput. Chem.* **2006**, *27*, 203.
[3] J. P. Ulmschneider, M. B. Ulmschneider, A. Di Nola, *J. Phys. Chem. B* **2006**, *110*, 16733.
[4] S. Bottaro, W. Boomsma, K. E. Johansson, C. Andreetta, T. Hamelryck, J. Ferkinghoff-Borg, *J. Chem. Theory Comput.* **2012**, *8*, 695.
[5] M. Habeck, M. Nilges, W. Rieping, *Phys. Rev. Lett.* **2005**, *94*, 18105.
[6] M. R. Betancourt, *J. Chem. Phys.* **2005**, *123*, 174905.
[7] C. Smith, T. Kortemme, *J. Mol. Biol.* **2008**, *380*, 742.
[8] J. P. Ulmschneider, W. L. Jorgensen, *J. Chem. Phys.* **2003**, *118*, 4261.
[9] G. Favrin, A. Irbäck, F. Sjunnesson, *J. Chem. Phys.* **2001**, *114*, 8154.
[10] R. L. Dunbrack, F. E. Cohen, *Protein Sci.* **1997**, *6*, 1661.
[11] A. Leaver-Fay, M. Tyka, S. M. Lewis, O. F. Lange, J. Thompson, R. Jacak, K. Kaufman, P. D. Renfrew, C. A. Smith, W. Sheffler, I. W. Davis, S. Cooper, A. Treuille, D. J. Mandell, F. Richter, Y.-E. A. Ban, S. J. Fleishman, J. E. Corn, D. E. Kim, S. Lyskov, M. Berrondo, S. Mentzer, Z. Popovíc, J. J. Havranek, J. Karanicolas, R. Das, J. Meiler, T. Kortemme, J. J. Gray, B. Kuhlman, D. Baker and P. Bradley, *Methods Enzymol.* **2011**, *487*, 545.
[12] T. Przytycka, *Proteins* **2004**, *57*, 338.
[13] T. Hamelryck, J. T. Kent, A. Krogh, *Plos Comput. Biol.* **2006**, *2*, e131.
[14] W. Boomsma, K. V. Mardia, C. C. Taylor, J. Ferkinghoff-Borg, A. Krogh, T. Hamelryck, *Proc. Natl. Acad. Sci. U.S.A.* **2008**, *105*, 8932.
[15] T. Harder, W. Boomsma, M. Paluszewski, J. Frellsen, K. E. Johansson, T. Hamelryck, *BMC Bioinform.* **2010**, *11*, 306.
[16] N. Gō, H. A. Scheraga, *Macromolecules* **1970**, *3*, 178.
[17] A. Irbäck, S. Mitternacht, S. Mohanty, *PMC Biophys.* **2009**, *2*, 2.
[18] G. A. Kaminski, R. A. Friesner, J. Tirado-Rives, W. L. Jorgensen, *J. Phys. Chem. B* **2001**, *105*, 6474.
[19] D. Qiu, P. S. Shenkin, F. P. Hollinger, W. C. Still, *J. Phys. Chem. A* **1997**, *101*, 3005.
[20] I. Lotan, F. Schwarzer, D. Halperin, J.C. Latombe, *J. Comput. Biol.* **2004**, *11*, 902.
[21] A. Irbäck, S. Mitternacht, *Proteins* **2007**, *71*, 207.
[22] D. W. Li, S. Mohanty, A. Irbäck, S. Huo, *PLoS Comput. Biol.* **2008**, *4*, e1000238.
[23] J. W. Ponder, F. M. Richards, *J. Am. Chem. Soc.* **1987**, *8*, 1016.
[24] B. Roux, T. Simonson, *Biophys. Chem.* **1999**, *78*, 1.
[25] W. C. Still, A. Tempczyk, R. C. Hawley, T. Hendrickson, *J. Am. Chem. Soc.* **1990**, *112*, 6127.
[26] B. Hess, C. Kutzner, D. van der Spoel, E. Lindahl, *J. Chem. Theory Comput.* **2008**, *4*, 435.
[27] K. E. Johansson, T. Hamelryck, *Proteins*, doi: 10.1002/prot.24277.
[28] U. H. E. Hansmann, Y. Okamoto, *J. Comput. Chem.* **1997**, *18*, 920.
[29] B. A. Berg, T. Neuhaus, *Phys. Rev. Lett.* **1992**, *68*, 9.
[30] B. Hesselbo, R. B. Stinchcombe, *Phys. Rev. Lett.* **1995**, *74*, 2151.
[31] J. Ferkinghoff-Borg, *J. Eur. Phys. J. B* **2002**, *29*, 481.
[32] E. De Alba, J. Santoro, M. Rico, M. Jimenez, *Protein Sci.* **1999**, *8*, 854.
[33] Schrödinger, LLC., The PyMOL Molecular Graphics System, Version 0.99rc6.
[34] T. Harder, M. Borg, W. Boomsma, P. Røgen, T. Hamelryck, *Bioinformatics* **2012**, *28*, 510.
[35] T. Hamelryck, M. Borg, M. Paluszewski, J. Paulsen, J. Frellsen, C. Andreetta, W. Boomsma, S. Bottaro, J. Ferkinghoff-Borg, *PLoS ONE* **2010**, *5*, e13714.
[36] K. Stovgaard, C. Andreetta, J. Ferkinghoff-Borg, T. Hamelryck, *BMC Bioinform.* **2010**, *11*, 429.
[37] N. G. Sgourakis, O. F. Lange, F. DiMaio, I. André, N. C. Fitzkee, P. Rossi, G. T. Montelione, A. Bax, D. Baker, *J. Am. Chem. Soc.* **2011**, *133*, 6288.
[38] A. Cavalli, X. Salvatella, C. M. Dobson, M. Vendruscolo, *Proc. Natl. Acad. Sci. U.S.A.* **2007**, *104*, 9615.
[39] Y. Shen, O. Lange, F. Delaglio, P. Rossi, J. M. Aramini, G. Liu, A. Eletsky, Y. Wu, K. K. Singarapu, A. Lemak, A. Ignatchenko, C. H. Arrowsmith, T. Szyperski, G. T. Montelione, D. Baker, A. Bax, *Proc. Natl. Acad. Sci. USA* **2008**, *105*, 4685.
[40] S. Olsson, W. Boomsma, J. Frellsen, S. Bottaro, T. Harder, J. Ferkinghoff-Borg, T. Hamelryck, *J. Magn. Reson.* **2011**, *213*, 182.
[41] S. Lloyd, *IEEE Trans. Inf. Theory* **1982**, *28*, 129.
[42] P. Røgen, B. Fain, *Proc. Natl. Acad. Sci. USA* **2003**, *100*, 119.
[43] T. Harder, M. Borg, S. Bottaro, W. Boomsma, S. Olsson, J. Ferkinghoff-Borg, T. Hamelryck, *Structure* **2012**, *20*, 1028.
[44] R. Das, D. Baker, *Annu. Rev. Biochem.* **2008**, *77*, 363.
[45] A. Irbäck, S. Mohanty, *J. Comput. Chem.* **2006**, *27*, 1548.

Received: 6 December 2012
Revised: 14 March 2013
Accepted: 20 March 2013
Published online on 26 April 2013

Protein Structure Validation and Refinement Using Amide Proton Chemical Shifts Derived from Quantum Mechanics

Anders S. Christensen, Troels E. Linnet, Mikael Borg, Wouter Boomsma, Kresten Lindorff-Larsen, Thomas Hamelryck, Jan H. Jensen (2013) Protein Structure Validation and Refinement Using Amide Proton Chemical Shifts Derived from Quantum Mechanics. *PLoS ONE* 8:e84123.

OPEN ACCESS Freely available online

PLOS ONE

Protein Structure Validation and Refinement Using Amide Proton Chemical Shifts Derived from Quantum Mechanics

Anders S. Christensen[1]*, Troels E. Linnet[2], Mikael Borg[3], Wouter Boomsma[2], Kresten Lindorff-Larsen[2], Thomas Hamelryck[3], Jan H. Jensen[1]

1 Department of Chemistry, University of Copenhagen, Copenhagen, Denmark, **2** Structural Biology and NMR Laboratory, Department of Biology, University of Copenhagen, Copenhagen, Denmark, **3** Structural Bioinformatics Group, Section for Computational and RNA Biology, Department of Biology, University of Copenhagen, Copenhagen, Denmark

Abstract

We present the ProCS method for the rapid and accurate prediction of protein backbone amide proton chemical shifts - sensitive probes of the geometry of key hydrogen bonds that determine protein structure. ProCS is parameterized against quantum mechanical (QM) calculations and reproduces high level QM results obtained for a small protein with an RMSD of 0.25 ppm ($r = 0.94$). ProCS is interfaced with the PHAISTOS protein simulation program and is used to infer statistical protein ensembles that reflect experimentally measured amide proton chemical shift values. Such chemical shift-based structural refinements, starting from high-resolution X-ray structures of Protein G, ubiquitin, and SMN Tudor Domain, result in average chemical shifts, hydrogen bond geometries, and trans-hydrogen bond ($^{h3}J_{NC'}$) spin-spin coupling constants that are in excellent agreement with experiment. We show that the structural sensitivity of the QM-based amide proton chemical shift predictions is needed to obtain this agreement. The ProCS method thus offers a powerful new tool for refining the structures of hydrogen bonding networks to high accuracy with many potential applications such as protein flexibility in ligand binding.

Citation: Christensen AS, Linnet TE, Borg M, Boomsma W, Lindorff-Larsen K, et al. (2013) Protein Structure Validation and Refinement Using Amide Proton Chemical Shifts Derived from Quantum Mechanics. PLoS ONE 8(12): e84123. doi:10.1371/journal.pone.0084123

Editor: Freddie Salsbury, Wake Forest University, United States of America

Received July 24, 2013; **Accepted** November 11, 2013; **Published** December 31, 2013

Funding: ASC is funded by the Novo Nordisk STAR PhD program. MB is funded by the Danish Council for Independent Research (FTP, 09-066546). WB and KL-L are supported by a Hallas-Møller stipend (to KL-L) from the Novo Nordisk Foundation. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have read the journal's policy and have the following conflicts: The authors declare funding from a commercial source, Novo Nordisk. This does not alter the authors' adherence to all the PLOS ONE policies on sharing data and materials.

* E-mail: andersx@nano.ku.dk

Introduction

Chemical shifts hold valuable structural information that is being used increasingly in the determination of protein structure and dynamics[1]. This is made possible primarily by empirical chemical shift predictors such as SHIFTS, SPARTA, SHIFTX, PROSHIFT, and CamShift [2–7]. While these methods generally offer quite accurate predictions, the predicted chemical shifts of backbone amide protons (δ_H) tend to be significantly less accurate than, for example, the proton on the α-carbon [8,9]. This is unfortunate since ^{15}N-HSQC forms a large fraction of all protein NMR studies and δ_H holds valuable information about the hydrogen bond geometry of the ubiquitous amide-amide hydrogen bonds that are key to protein secondary structure. Parker, Houk and Jensen [10] have proposed a δ_H-predictor that was shown to offer significantly more accurate predictions, although this was only demonstrated for 13 δ_H-values. The method suggests that there is an exponential dependence of δ_H in the NH··O = C bond length (as suggested by Barfield [11] and Cornilescu *et al.* [12]) as well as a non-negligible contribution from cooperative effects in hydrogen bonding networks. This exponential dependence makes empirical parameterizations of δ_H-predictors challenging since even small discrepancies between the structure used in the parameterization (usually an X-ray structure without explicitly represented hydrogens) and the solution-phase structural ensemble that gives rise to the experimentally observed δ_H-values can have a significant effect. The method by Parker *et al.* addresses this problem by parameterization against δ_H-values obtained by quantum mechanical (QM) calculations, and is similar in spirit to the QM-based α-carbon chemical shift predictor CheShift developed by Vila *et al.* [13,14]. Both studies noted that the QM-based chemical shift predictors tend to be more sensitive to small structural changes compared to popular empirical chemical shift predictors and therefore promises to be valuable tools in protein structure validation and refinement. Here we present several key advances in the use of backbone amide proton chemical shifts to refine and validate the geometry of the amide-amide hydrogen bonding network in proteins. First we present and validate the ProCS method which extends the QM-based backbone amide proton chemical shift predictor proposed by Parker *et al.* [10]. Second we present a computational methodology for using ProCS and experimental δ_H-values to refine the hydrogen bond-geometries of proteins. This is accomplished by implementing ProCS in the Markov chain Monte Carlo (MCMC) protein simulation framework PHAISTOS [15], and using this in

combination with a molecular mechanics (MM) force field. Third, we show for a number of small proteins that structural refinement against experimental δ_H values using ProCS leads to hydrogen bond geometries that are in closer agreement with high-resolution X-ray structures and experimental trans-hydrogen bond spin-spin coupling constants ($^{h3}J_{NC'}$) compared to using an energy function based on the empirical chemical shift predictor CamShift [7] or solely using a force field (OPLS-AA/L [16] with the GB/SA continuum solvent model [17]).

Results and Discussion

The ProCS method

The ProCS program uses a modified implementation of the formula developed by Parker *et al.*[10] where the amide proton chemical shift is approximated by a sum of additive terms:

$$\delta_H = \delta_{BB} + \Delta\delta_{1^{\circ}HB} + \Delta\delta_{2^{\circ}HB} + \Delta\delta_{3^{\circ}HB} + \Delta\delta_{RC} \quad (1)$$

Here, δ_{BB} is a backbone term that depends on the (ϕ,ψ) torsion angles of the residue, $\Delta\delta_{1^{\circ}HB}$ is due to a primary hydrogen bond directly to the amide proton in question, $\Delta\delta_{2^{\circ}HB}$ is due to a secondary hydrogen bond to the carbonyl oxygen in the amide group, $\Delta\delta_{3^{\circ}HB}$ is a small term that incorporates further polarization due to hydrogen bonding at the primary and/or secondary bonding partner and $rDelta\delta_{RC}$ describes magnetic perturbations due to ring currents in nearby aromatic side chains. ProCS calculates amide proton chemical shift values referenced to dimethyl-silapentane-sulfonate (DSS).

We have replaced the original δ_{BB} term, which was a crude 3-step function, by a scaled version of the (ϕ,ψ) backbone torsion angle hypersurface parametrized by Czinki and Császár [18]. The δ_{BB} term is given as

$$\delta_{BB} = 0.828 \cdot (\mathrm{ICS}(\phi,\psi) + 0.77 \text{ ppm}) \quad (2)$$

where $\mathrm{ICS}(\phi,\psi)$ is the n-th order cosine series given in reference [18]. The scaling is necessary to account for differences in choice of basis set and molecular geometry optimization [19].

In the cases described by Parker *et al.*, $\Delta\delta_{RC}$-values are obtained through the SHIFTS web-interface[3]. Since this would be impractical, we implemented the point-dipole [20,21] approximation given by:

$$\Delta\delta_{RC} = i\ B\ \frac{1 - 3\cos^2(\theta)}{|\vec{r}|^3} \quad (3)$$

where i is an intensity parameter which depends on the type of aromatic ring, B is a constant of 30.42 ppm Å^3, $\vec{r}$ is the vector between the amide proton and the center of the aromatic ring and θ is the angle between $\vec{r}$ and the normal to the plane of the aromatic ring located on its center. The values of i and B are obtained from the parameter set by Christensen *et al.* [22].

The following expression for $\Delta\delta_{1^{\circ}HB}$ was implemented for primary bonds to backbone amide carbonyl oxygen atoms:

$$\Delta\delta_{1^{\circ}HB} = [4.81\cos^2(\theta) + \sin^2(\theta)\{3.10\cos^2(\rho)$$

$$-0.84\cos(\rho) + 1.75\}]e^{-2.0\ \mathrm{A}^{-1}(r_{OH} - 212.760\ 2\mathrm{A})} \cdot 1 \text{ ppm} \quad (4)$$

This formula originates from the works of Barfield[11] and is fitted to chemical shifts computed for model systems of hydrogen bonding between two formamide molecules. In order to treat hydrogen bonding to other oxygen atom types (carboxylic acids and alcohols as found in side chains and C-terminal), we carried out similar scans (see Section S2 and Fig. S4 in Supporting Information S1) over bond angles and lengths and stored these in lookup-tables from which the chemical shift perturbation due to any hydrogen bonding geometry can be interpolated. Hydrogen bonding to carboxylic acid oxygen atoms interaction were modeled by *N*-methylacetamide/acetate dimers, while bonds to alcohols oxygen atoms were modeled by *N*-methylacetamide/methanol dimers.

For non-hydrogen bonding amide protons, which are found primarily on the protein surface, $\Delta\delta_{1^{\circ}HB}$ is approximated as the interaction between a water molecule and an *N*-methylacetamide molecule. In this case, $\Delta\delta_{1^{\circ}HB}$ is equal to 2.07 ppm for an energy minimized bonding geometry (see Section S3 and Fig. S5 in Supporting Information S1). The functional forms of $\Delta\delta_{2^{\circ}HB}$ and $\Delta\delta_{3^{\circ}HB}$ were kept as described in reference [10].

Reproducing QM chemical shifts

ProCS predictions result from several terms [Eq. 1] that are assumed to be additive. To test this additivity assumption we use density functional theory (DFT) and compute chemical shielding values (at the B3LYP/cc-pVTZ/PCM level) for the crystal structure of human parathyroid hormone, residues 1–34 at 0.9 Å resolution, PDB-code 1ET1 [23]. Chemical shift values for amide protons at the termini are excluded from the statistics presented in this section, since they do not participate in any hydrogen bonds in the crystal structure. Using the linear scaling method due to Jain *et al.* [24] similar DFT calculations reproduce experimental proton chemical shifts of a test set of 80 small to medium sized molecules to an RMSD of 0.13 ppm. [24]

ProCS reproduces the QM calculation with an RMSD of 0.25 ppm (Table 1) based on the same structure. ProCS is parameterized based on a number of DFT calculations (see Methods section) which have been shown to yield proton chemical shifts within 0.16 ppm of experimental values for small organic molecules [19]. Thus, the error from non-additivity is roughly the same as the expected deviation from experiment.

The chemical shifts predicted by empirical methods do not agree well with the DFT results, with RMSD values ranging from 0.56 to 0.70 ppm (see Table 1 and Fig. 1). The DFT chemical shifts span a relatively large range (5.8–9.3 ppm) while the empirically predicted chemical shifts span a very narrow range (up to 6.9–8.9 ppm for SPARTA+) - see Fig. 1. This indicates that the empirical methods are less sensitive to small differences in hydrogen bond geometry found in the X-ray structure.

Reproducing experimental chemical shifts from X-ray structures

The QM method used here reproduces small molecule ^{1}H chemical shifts with an RMSD of 0.13 ppm [24]. The RMSD between the chemical shifts calculated by QM using the static X-Ray structure and the experimental data obtained in solution is 0.66 ppm. The main sources of this discrepancy are likely inaccuracies in the hydrogen bond lengths in the X-ray structure compared to solution, since there is an exponential dependence of the proton chemical shifts on this distance [Eq. 4], and/or the use of a single structure rather than a structural ensemble.

The corresponding RMSD to experimental data for ProCS (0.63 ppm) is similar to the QM RMSD and significantly larger than the 0.25 ppm RMSD between QM and ProCS, indicating

Table 1. Correlation coefficients and RMSD between five chemical shift predictors, chemical shifts derived from quantum mechanics (B3LYP/cc-pVTZ/PCM) chemical shifts and experimental values.

Data source[a]	Exp'tl	Exp'tl	QM	QM
	r	RMSD	r	RMSD
ProCS	0.54	0.63	0.94	0.25
SHIFTS[2]	0.64	0.37	0.59	0.70
SHIFTX[5]	0.69	0.37	0.71	0.62
SPARTA+[40]	0.69	0.42	0.68	0.56
CamShift[7]	0.64	0.32	0.59	0.66

[a]The crystal structure of human parathyroid hormone, residues 1–34 at 0.9 Å resolution (PDB-code 1ET1[23]) is used as input structure in all chemical shift calculations.
doi:10.1371/journal.pone.0084123.t001

Table 2. Reproduction of experimental amide proton chemical shift values based on 13 X-ray structures with a crystallographic resolution of 1.35 Å or less.

Method	$\langle r \rangle$[a]	$\langle$RMSD$\rangle$[b]
ProCS	0.58	1.13 ppm
SHIFTS[2]	0.56	0.64 ppm
SHIFTX[5]	0.71[c]	0.51 ppmc
SPARTA+[40]	0.79	0.40 ppm
CamShift[7]	0.74	0.46 ppm

$\langle r \rangle$">denotes the average correlation coefficient over the 13 structure.
⟨">RMSD⟩ denotes the average root mean square deviation over the 13 structure.
[c]For SHIFTX, three structures displayed over fitting behavior with $r \approx 0.99$. These structures are excluded from the average values.
doi:10.1371/journal.pone.0084123.t002

that ProCS is sufficiently accurate to identify inaccuracies in the X-ray structure, and/or the effect of using a single structure rather than a structural ensemble. A similar comparison to experiment for 13 other proteins is given in Table 2 (PDB-codes: 1BRF, 1CEX, 1CY5, 1ET1, 1I27, 1IFC, 1IGD, 1OGW, 1PLC, 1RGE, 1RUV, 3LZT, 5PTI). The deviation from experiment for the empirical methods are significantly smaller than for ProCS with RMSD values ranging from 0.46 to 0.64 ppm (Table 2). A likely explanation for this is that the empirical methods are parameterized using X-ray structures. In order for these methods to produce low RMSD values relative to experiment they need to be insensitive to errors in protein structure.

Refining protein structures based on chemical shifts

If indeed the difference in experimental and computed chemical shifts reports on inaccuracies in the protein structure, then minimizing this difference can be used for structural refinement. To test this hypothesis we generate structural ensembles that minimizes the difference in computed and observed chemical shifts to the specified uncertainty in the chemical shift model and determine the quality of these structures by comparison to experimental structures and coupling constants (next section).

Refinement is accomplished using a Markov chain Monte Carlo (MCMC) technique described in detail in the Methods section. In short, the method involves Monte Carlo sampling of structural changes using a posterior distribution constructed using the OPLS-AA/L fore field [16] with the GB/SA implicit solvent model [17] (referred to hereafter simply as "OPLS") and amide proton chemical shifts differences from experiment computed using either CamShift or ProCS. We note that the resulting ensemble is not a dynamic ensemble but an ensemble that reflects experimentally measured amide proton chemical shifts. The simulation lengths are roughly equivalent to 6–10 ns of molecular dynamics simulations [25]. We refine the structure of ubiquitin, Protein G, and SMN Tudor domain each based on three energy functions: OPLS alone, OPLS+ProCS and OPLS+CamShift. Each MC refinement results in an ensemble of 24,000 structural samples for Ubiquitin and 40,000 for Protein G and SMN Tudor Domain, from which average chemical shifts for each amide proton are computed. The results are summarized in Table 3.

The average ProCS chemical shifts are in better agreement with experiment (RMSD 0.81 ppm) compared to using X-ray structures (RMSD 1.10 ppm). The respective RMSD values for amide protons hydrogen bonded to backbone amide groups, other hydrogen bonds, and no hydrogens bonds are 0.31 ppm, 0.78 ppm and 1.09, respectively. These RMSD values reflect the uncertainties defined for each kind of hydrogen bonding situation in the ProCS model (see Methods section) meaning that the simulations have indeed converged to a distribution of structures reflecting the experimental chemical shifts within the accuracy of the ProCS model at the given temperature. A corresponding structural ensemble generated solely from the OPLS force field increases the RMSD from experiment to 1.52 ppm, indicating more inaccurate hydrogen bond geometries (more on this in the next section).

An MC-based structural refinement based on OPLS and chemical shifts derived from CamShift has no substantial effect

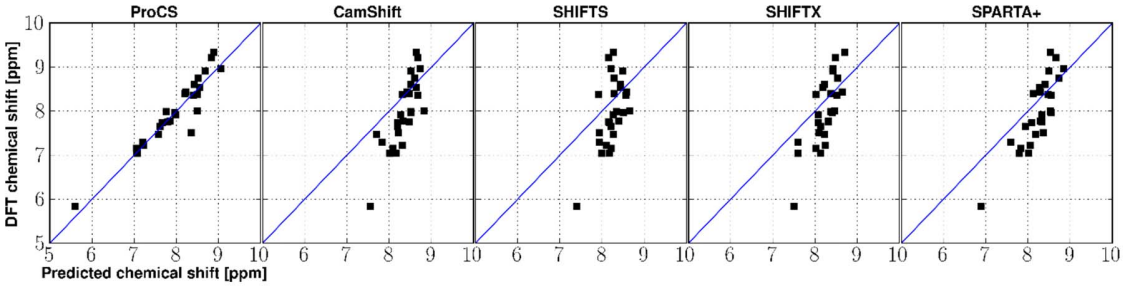


Figure 1. Correlation between chemical shift predictions from five different NMR prediction methods and quantum mechanical chemical shifts for human parathyroid hormone, residues 1–37 (PDB code: 1ET1). Blue lines represent a 1-to-1 correlation.
doi:10.1371/journal.pone.0084123.g001

Table 3. Statistics for three different types of protein simulations.

	ProCS	CamShift	⟨Bond length	
Structures[a]	^{1}H RMSD	^{1}H RMSD	deviation⟩[b]	$^{h3}J_{NC'}$ RMSD
Ubiquitin Ensembles: CamShift + OPLS	0.79 ppm	-	0.03 Å	0.17 Hz
Ubiquitin Ensembles: CamShift + OPLS	-	0.50 ppm	0.37 Å	0.17 Hz
Ubiquitin Ensembles: OPLS (no chemical shifts)	1.56 ppm	0.60 ppm	0.41 Å	0.18 Hz
1UBQ X-ray starting structure	1.22 ppm	0.51 ppm	-	0.22 Hz
SMN Tudor Domain Ensembles: ProCS + OPLS	0.93 ppm	-	0.09 Å	0.24 Hz
SMN Tudor Domain Ensembles: CamShift + OPLS	-	0.46 ppm	0.17 Å	0.23 Hz
SMN Tudor Domain Ensembles: OPLS (no chemical shifts)	1.47 ppm	0.61 ppm	0.22 Å	0.23 Hz
1MHN X-ray starting structure	1.09 ppm	0.65 ppm	-	0.24 Hz
Protein G Ensembles: ProCS + OPLS	0.69 ppm	-	0.06 Å	0.14 Hz
Protein G Ensembles: CamShift + OPLS	-	0.52 ppm	0.38 Å	0.18 Hz
Protein G Ensembles: OPLS (no chemical shifts)	1.54 ppm	0.68 ppm	0.37 Å	0.20 Hz
1PGB X-ray starting structure	1.21 ppm	0.55 ppm	-	0.17 Hz

[a]The ensembles are obtained from MCMC simulations using either OPLS-AA/L with the GB/SA solvent model (OPLS) force field energy or OPLS energy plus a chemical shift energy term from from either ProCS or CamShift. Values are calculated over four runs on each of three protein structures, Ubiquitin, Protein G and SMN Tudor Domain, or their static X-ray structure.
[b]The mean bond length deviation denotes the mean absolute difference between the mean hydrogen bond length observed in the sampled structures to the mean hydrogen bond length observed in the corresponding X-ray structure noted below.
doi:10.1371/journal.pone.0084123.t003

on the chemical shift RMSD compared to the X-ray structure (0.50 vs 0.46 ppm). Using the OPLS-derived structural ensemble increases the RMSD by 0.1 ppm compared to using X-ray structures when CamShift is used to calculate chemical shifts. This indicates that an OPLS-based refinement does not improve the hydrogen bonding geometry and that CamShift is less sensitive to a change in structure compared to ProCS.

Hydrogen bond geometries

The H··O distances and H··O = C angles of the backbone amide-amide hydrogen bonds for which $^{h3}J_{NC'}$ coupling constants have been measured (see next section) are extracted from the ensembles and compared to the corresponding values found in the experimental X-ray structures with hydrogens added from PDB2PQR [26,27]. The result are shown in Table 3 and Figures 2 and 3.

Fig. 2 shows the distributions of H··O distances from the ensembles computed using the three energy terms described in the previous section. Structural refinement using OPLS and ProCS for ubiquitin results in ensembles with average H··O distances that have an RMSD within 0.02 Å of those found in the X-ray structures 1UBQ and 1UBI (both 1.80 Å X-ray resolution) and 0.04 Å from the ubiquitin structure 1OGW (1.30 Å X-ray resolution) in which the leucine residues 50 and 67 have been replaced by fluoro leucine. For Protein G we note that the resulting ensemble does not have an average H··O distance that agrees well (0.07 Å difference) with the starting structure 1PGB (1.92 Å X-ray resolution). However the difference from the 1PGA structure (2.07 Å X-ray resolution) and the more accurate 1IGD structure (X-ray resolution of 1.1 Å) is much less, 0.02 Å and 0.00 Å, respectively. The 1IGD structure is a close homologue which has 89% sequence identity score and 95% sequence similarity. In the case of the SMN Tudor Domain, ProCS-based refinement results in slightly longer amide-amide hydrogen bond lengths (0.02 Å on average) compared to the X-ray structure 1MHN.

In contrast, structural refinement using CamShift and OPLS or just OPLS leads to increases in average H··O bond lengths of up to 0.15 Å, with a standard deviation 2–3 times larger than that found in the OPLS+ProCS simulation. In all cases use of CamShift has relatively little effect on the ensemble average H··O distance compared to just using OPLS.

In all cases, the use of ProCS leads to a significantly smaller standard deviation in H··O bond lengths: 0.017 Å compared to 0.045 and 0.041 Å for CamShift+OPLS and OPLS, respectively (Fig. 3A). The H··O = C bond angles observed in the ProCS+OPLS simulations are on average within -2.0° of corresponding value observed in the X-ray structures. The same bond angle differences are -6.7° and -7.4° observed in the CamShift+OPLS and OPLS simulations, respectively (Fig. 3B).

Trans-hydrogen bond coupling constants

Better agreement with X-ray structures does not necessarily imply better solution-phase structures. In order to compare the resulting ensembles to solution-phase data we compute average trans-hydrogen bond coupling constants and compare these to experimental values. Experimental trans-hydrogen bond $^{h3}J_{NC'}$ spin-spin coupling constants represent a very sensitive measure for solution-phase hydrogen bonding conformations and are known to correlate with amide proton chemical shifts [28]. The coupling constants depend exponentially on the hydrogen bonding distance and on bond angles [11]. Data from ensemble back-calculated $^{h3}J_{NC'}$ spin-spin coupling constants are summarized in Fig. 4 and Table 3.

In the ubiquitin simulations, the OPLS force field on its own does not yield ensemble $^{h3}J_{NC'}$ averages in good agreement with experimental data. In this simulation, several hydrogen bonds were eventually broken. Calculated $^{h3}J_{NC'}$-values for these partly unfolded hydrogen bonds show up close to 0 Hz (see Fig. 4A). The RMSD to experimental values is here 0.18 Hz. Adding the energy term from amide proton chemical shifts via CamShift does not help keeping these hydrogen bonds fixed, but results in a minor improvement in RMSD to 0.17 Hz. Adding the amide proton

chemical shifts energy term via ProCS to the OPLS force field stabilized the hydrogen bonds and also gave an improvement in the RMSD values to 0.14 Hz, which is close to that of the most accurate structural NMR ensembles of ubiquitin (see Table 4). For Protein G we obtained similar RMSD values: 0.20 Hz, 0.14 Hz and 0.18 Hz for the OPLS alone, OPLS+ProCS and the OPLS+CamShift simulations, respectively. In the SMN Tudor Domain simulation, the average $^{h3}J_{NC'}$ value of all three types of simulations were comparably close to experimental values 0.24, 0.24 and 0.23 Hz for OPLS alone, OPLS+ProCS and the OPLS+CamShift simulations, respectively. Thus, overall the coupling constants based on the ProCS refined ensembles are indeed in better agreement with experimental values indicating the refinement led to improved hydrogen bond geometries compared to using OPLS or OPLS+CamShift.

Impact on Q-factor

In this section we investigate how amide proton chemical shifts restraints affect back-calculated $^1D_{NH}$ residual dipolar couplings (RDCs) compared to experimental values for ubiquitin. RDCs are attractive in this regard since they report on structural features that are not related to hydrogen bonding conformations as studied intensively in the previous sections. The Q-factor is a qualitative measure for the agreement between back-calculated RDCs and the corresponding experimentally observed values [29].

We find, that for our Ubiquitin ensemble generated using the OPLS force field alone has a Q-factor of 0.29 while inclusion of chemical shifts only gives a very modest improvement of this figure to 0.27 for both CamShift and ProCS as chemical shift model. The same value calculated for the three X-ray structures 1UBQ, 1UBI and 1OGW are 0.22, 0.25 and 0.26, respectively. For six NMR-based ensembles the Q-factor is in the range 0.04–0.38, though in some cases the ensembles were refined against the RDCs (see Table 4). We observe no significant correlation ($P<0.05$) between RMSDs for predicted chemical shifts or spin-spin couplings constant to their experimental values and the calculated Q-factor for the 12 cases presented in Table 4.

While amide proton chemical shifts have some dependence on the dihedral angles of the backbone, the dependence on the particular hydrogen bonding conformations is much larger in comparison. This is due to an exponential dependence on the hydrogen bond length.

The distribution from which we sample chemical shifts is constructed from a prior distribution based on the OPLS force field and a likelihood which contains information from experimental chemical shifts. We expect that structural features of the resulting ensemble, which are not local to the hydrogen bond geometry, will largely reflect the prior distribution, i.e. in this our case, the OPLS force field.

Computational efficiency

Executing the simulations on one core of a Intel Xeon X5560 running at 2.80 GHz with the 1UBQ structure, the average evaluation time of the three different energy-terms were OPLS-AA/L: 27 ms, CamShift 1.35: 4.7 ms, ProCS: 0.74 ms. Similar evaluation times were observed for the 1MHN and 1PGB simulations. Note that, in our implementation, the CamShift term calculates chemical shifts for six atoms per residue, even if those chemical shifts are not a used to evaluate the corresponding energy term. The OPLS and CamShift terms were implemented with a caching algorithm, so only the subset of parts of the chemical shift terms that change after a local Monte Carlo move were recomputed. This approach was not implemented for ProCS since the OPLS force field energy evaluation is by far the most computationally expensive step. Running on four cores, we obtained between 10 to 16 mio Monte Carlo iteration steps total *per* day, depending on the protein size and combination of energy terms.

Methods

Monte Carlo refinement of protein structure

We employ Markov chain Monte Carlo sampling from a Bayesian posterior distribution to perform protein structure refinements and simulations. MCMC simulations are attractive because no gradient expressions need to be derived for ProCS. Bayesian inference[30] provides a rigorous mathematical framework for the inference of protein structure from experimental data. It involves the construction of a posterior distribution, which consists of a prior distribution and a likelihood. The former brings in general information on protein structure, and in our case is

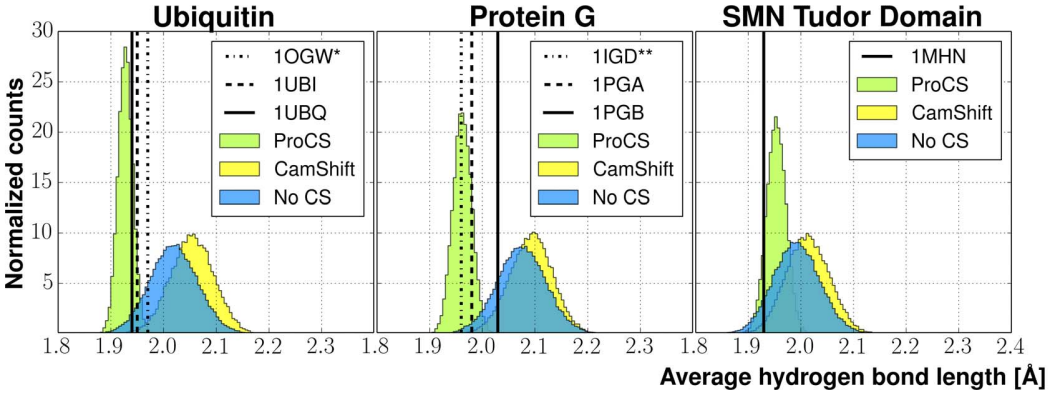


Figure 2. Distribution of average hydrogen bond lengths throughout Monte Carlo simulations on Ubiquitin, Protein G and SMN Tudor Domain. Histograms are normalized (to an area of 1) to fit identical axes. Vertical lines indicate average values obtained from experimental X-ray structures (PDB-codes are noted in the figure legends). The blue histogram represents the simulation with only the molecular mechanics energy from the OPLS-AA/L force field with the GB/SA solvent model (but no chemical shift energy term). Green and yellow histograms indicate the use of OPLS force field plus an additional chemical shift energy term from ProCS or CamShift, respectively. *1OGW contains fluoro leucine at residues 50 and 67. **1IGD is a closely related homologue (see text).
doi:10.1371/journal.pone.0084123.g002

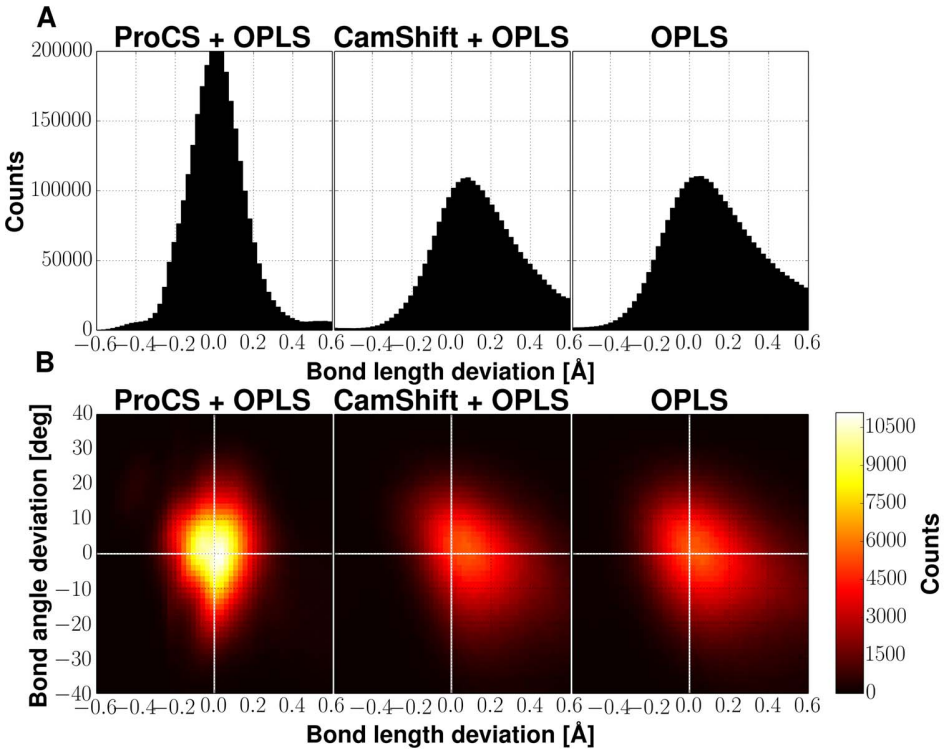


Figure 3. Deviation in hydrogen bonding geometries between the experimental X-ray structure and samples obtained from Markov Chain Monte Carlo (MCMC) simulations using the OPLS-AA/L force field with the GB/SA solvent model with either no chemical shift energy term or a chemical shift energy from either ProCS or CamShift. Data is calculated over all amide-amide bonding pairs for which experimental $^{h3}J_{NC'}$ spin-spin coupling constants were present. (A) shows the distribution of the deviations found in the MCMC ensembles from the experimental hydrogen bond length found in the X-ray structure. (B) shows the correlation of deviations in hydrogen bond lengths and H··O = C bond angles from the experimental X-ray structures.
doi:10.1371/journal.pone.0084123.g003

based on the OPLS energy function. The latter brings in the experimental data, and is based on the difference between the back-calculated data from a simulated structure and the experimental data. Using PHAISTOS, we draw samples from the joint probability distribution, which is given by:

$$p(X|\{\delta_i^{\exp}\},I) \propto p(\{\delta_i^{\exp}\}|X,I)p(X|I) \quad (5)$$

where X represents a protein structure, $\{\delta_i^{\exp}\}$ is experimental chemical shift data and I denotes prior information, such as sequence and knowledge about the uncertainties in the prediction model. The prior distribution $p(X|I)$ is proportional to $\exp(-\beta E_{FF})$, where E_{FF} is the molecular mechanics force field potential energy and $\beta = 1/k_B T$. $p(\{\delta_i^{\exp}\}|X,I)$ denotes the probability of observing experimental data given a trial structure. Under the assumption that the error in the chemical shift prediction model follows a Gaussian distribution with some set of standard deviations $\{\sigma_i\}$, the expression for $p(\{\delta_i^{\exp}\}|X,I)$ is:

$$p2(\{\delta_i^{2exp}\}\Big|X, \{\sigma_i\}) = \Pi_{i=1}^{n}\left[\sqrt{\frac{1}{2\pi\sigma_i^2}}\exp\left\{-\frac{(\Delta\delta_i)^2}{2\sigma_i^2}\right\}\right] \quad (6)$$

where $\Delta\delta_i$ is the discrepancy between predicted and experimental data for the i-th nucleus of the data set in the trial structure, X. This formulation of the posterior distribution assumes that the prior distribution on X is also a good prior distribution for the chemical shift differences, $\Delta\delta_i$, otherwise an additional term would be required[31]. The set of standard deviations, $\{\sigma_i\}$ was assigned based on the primary bond type, since, for instance, the model for solvent exposed amide protons is much cruder than the amide-amide bonding model. σ_i was set to 0.3 ppm, for primary bonds to another backbone amide, 0.5 ppm to a side chain amide group, 0.8 ppm to a side chain alcohol or carboxylic acid group and 1.2 ppm for solvent exposed amide protons and other types of bond not included in the prediction model.

Protein Structures and NMR data

All protein structures used in this study were downloaded from the RCSB Protein Data Bank[32] (PDB) and protonated using PDB2PQR 1.5, [26,27] with PROPKA[33] to determine protonation states at the pH at which NMR data was recorded. Chemical shift data were obtained from the RefDB[34] or the Biological Magnetic Resonance Bank[35], and subsequently re-referenced through Shiftcor[34]. $^{h3}J_{NC'}$ spin-spin coupling constants for 1PGB, 1UBQ and 1MHN were obtained from references [28], [12] and [36], respectively.

MCMC simulations

MCMC simulations were carried out in PHAISTOS v1.0-rc1 (rev. 335) using the Metropolis-Hastings algorithm at 300 K. The simulations are initialized from the experimental crystal structures. Four independent trajectories were simulated for each protein structure. A total of 100 mio MC steps were taken for each trajectory for Protein G and the SMN Tudor Domain simulation

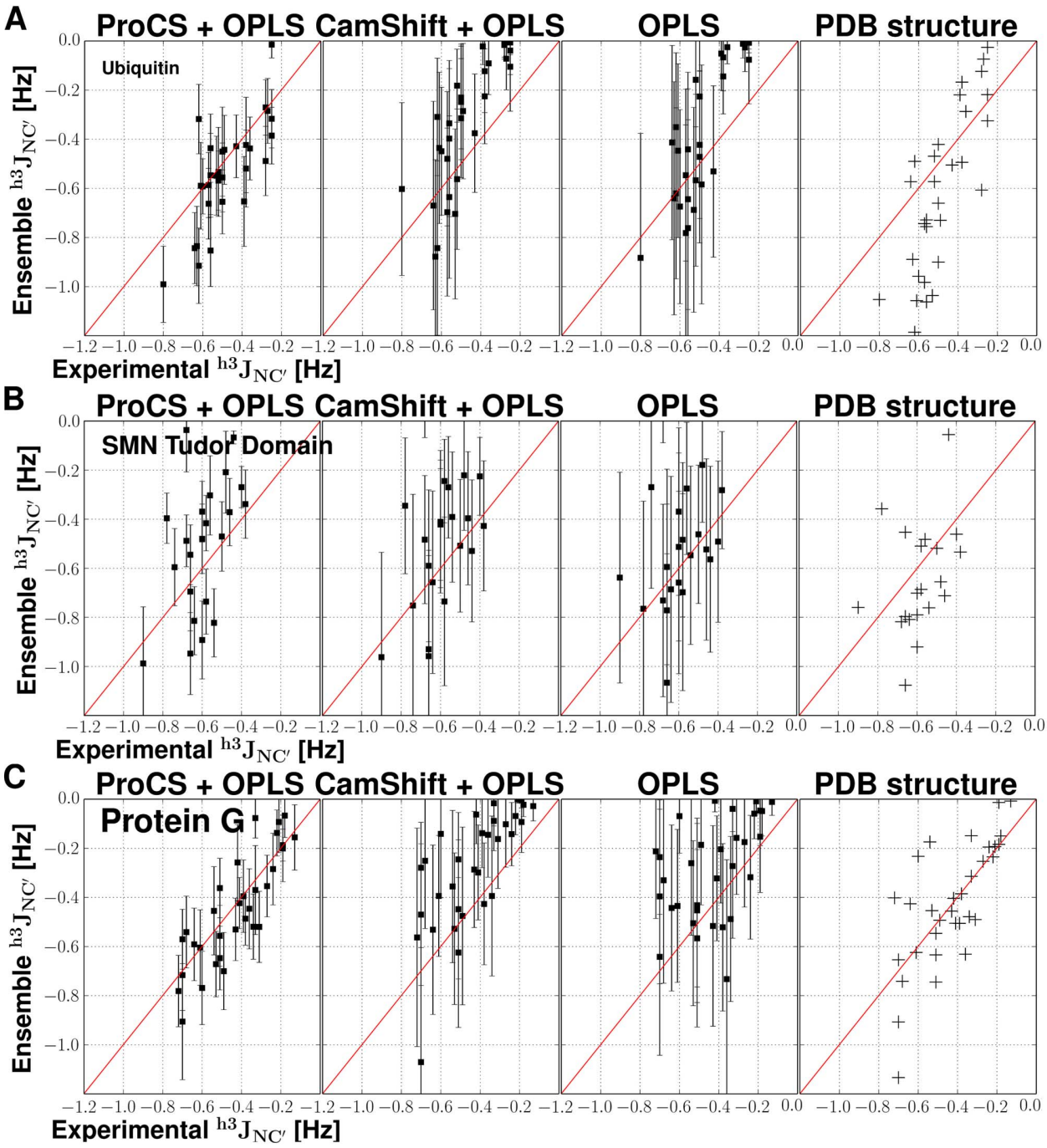


Figure 4. Reproducing experimental $^{h3}J_{NC'}$ spin-spin coupling constants via different structural ensembles and experimental X-ray structures. Squares denote the average coupling constant observed for that hydrogen bond in the ensemble and error bars represent the standard deviation observed throughout the simulations. Crosses represent the spin-spin coupling constants calculated using the static experimental X-ray structure. Results from simulations on ubiquitin is displayed in A, SMN Tudor domain in B and Protein G in C. Left column displays simulations only the OPLS-AA/L force field with the GB/SA solvent model (OPLS) and the ProCS energy term; second column is from OPLS plus the CamShift energy term; thrid column is for the simulation with only the OPLS force field energy. In the rightmost column $^{h3}J_{NC'}$ are computed from the corresponding X-ray structure.
doi:10.1371/journal.pone.0084123.g004

and 85 mio MC steps for the Ubiquitin simulation. Structures were saved every 10,000 Monte Carlo step. The Monte Carlo move-set was composed of 25% CRISP backbone moves[25] and 75% uniform side chain moves. The force field energy was calculated using the OPLS-AA/L force field [16] with the GB/SA continuum solvent model [17]. The following crystal structures obtained from the PDB were used as starting structures in the simulations: 1PGB (Protein G), 1UBQ (Ubiquitin) and 1MHN

Table 4. Statistics for selected ubiquitin ensembles and X-ray structures.[a]

	(CamShift)	(CamShift)	(ProCS)	(ProCS)	$^{h3}J_{NC'}$	
PDB-ID	^{1}H RMSD	r	^{1}H RMSD	r	RMSD	Q-factor
[b]2KOX	0.29	0.84	0.68	0.86	0.12	0.04
[c]2K39	0.34	0.82	0.98	0.77	0.13	0.07
[d]2KN5	0.23	0.91	0.71	0.82	0.12	0.22
[e]2NR2	0.44	0.74	1.35	0.64	0.14	0.25
[f]1XQQ	0.38	0.81	0.92	0.77	0.14	0.38
[g]1D3Z	0.41	0.79	1.00	0.71	0.30	0.06
[h]1UBQ	0.40	0.77	0.92	0.72	0.22	0.22
[i]1UBI	0.40	0.77	0.97	0.73	0.33	0.25
[j]1OGW	0.36	0.73	0.84	0.73	0.17	0.26
[k]OPLS + ProCS	0.32	0.79	0.17	0.98	0.14	0.27
[k]OPLS + CamShift	0.32	0.90	1.15	0.86	0.17	0.27
[k]OPLS	0.48	0.78	1.11	0.78	0.18	0.29

[a]Chemical shifts RMSD and r values are calculated for the residues for which $^{h3}J_{NC'}$ spin-spin coupling constants have been measured. [12]
[b]ERNST method/CHARMM27 + NOE + RDC [41]
[c]OPLS/AA-L + NOE + RDC [42]
[d]Backrub method/Rosetta all-atom energy + RDC [42]
[e]MUMO method/CHARMM22 + NOE + RDC [43]
[f]DER method/CHARMM22 + NOE + S^2 [44]
[g]NOE + RDC [45]
[h]X-ray 1.80 Å structure [46]
[i]X-ray 1.80 Å structure [47]
[j]X-ray 1.32 Å structure (synthetic protein with fluoro-LEU at residues 50 and 67) [48]
[k]The methods presented here
doi:10.1371/journal.pone.0084123.t004

(SMN Tudor Domain). Time evolution of Monte Carlo energy and chemical shift RMSDs are available in the Supplementary Information (Section S1, Figures S1–S3 of Supporting Information S1).

Back calculation of spin-spin coupling constants

$^{h3}J_{NC'}$ spin-spin coupling constants were calculated using the approximation by Barfield[11].

$$^{h3}\mathrm{J}_{\mathrm{NC}'}(\theta,\rho,\gamma_{OH}) = [-1.31\cos^2(\theta) + \{0.62\cos^2(\rho) + 0.92\cos(\rho) + 0.14\}\sin^2(\theta)]\ \mathrm{e}^{-3.2\mathrm{A}} - 1(\mathrm{r}_{2OH} - 1.760\mathrm{A}).1\ Hz \quad (7)$$

Here, the coupling depend on the $\angle$N-H··O = C angle, ρ, $\angle$H··O = C, θ, and the hydrogen bonding distance, r_{OH}. From the MCMC ensembles, the mean $^{h3}J_{NC'}$ spin-spin coupling constant was calculated via Eqn. 7 and the standard deviation was calculated as the root mean square deviation from the mean. The $^{h3}J_{NC'}$ RMSD to experiment is then given as

$$^{h3}\mathrm{J}_{NC'}RMSD = \sqrt{\frac{\sum_{\mathrm{i}}\left(^{h3}\mathbf{J}_{NC'}^{\exp,\mathrm{i}} - \langle ^{h3}\mathbf{J}_{NC'}^{calc,\mathrm{i}}\rangle\right)^2}{\mathrm{N}}} \quad (8)$$

where $\langle ^{\mathrm{h3}}J_{\mathrm{NC}'}^{\mathrm{calc},i}\rangle$ is the average value over the ensemble for the i'th coupling constant.

QM NMR calculations

All density functional theory (DFT) calculations of NMR isotropic shielding constants involved in the parametrization of ProCS were carried out in Gaussian 03[37]. Data was obtained at the GIAO/B3LYP/6-311++G(d,p)//B3LYP/6-31+G(d) level of theory using the scaling technique by Rablen *et al.* [19].

The NMR calculation on the 1ET1 protein structure was carried out at the B3LYP/cc-pVTZ/PCM level of theory with a water-like dielectric constant of 78.3553. In this case shielding constants were converted to chemical shifts using the scaling factor obtained by Jain *et al.* [24], assuming that the value of the dielectric constant has a negligible contribution to the scaling factors.

Calculation of ubiquitin Residual Dipolar Couplings

Residual dipolar couplings were back-calculated from the structural ensembles using singular value decomposition to fit the alignment tensor [38]. Ensemble averaging was taken into account so that all structures simultaneously were fitted to a single alignment tensor [39]. The agreement to experimental values was calculated via the Q-factor: [29]

$$\mathrm{Q} = \frac{\sqrt{\Sigma(RDC^{\exp} - RDC^{calc})^2}}{\sqrt{\Sigma(RDC^{calc})^2}} \quad (9)$$

Conclusions

ProCS is a QM-based backbone amide proton chemical shift (δ_{H}) predictor that can deliver QM quality chemical shift predictions for a protein structure in a millisecond. δ_{H}-values predicted using X-ray structures are in worse agreement with experiment, compared to those of the popular empirical chemical shift-predictors CamShift, SHIFTS, SHIFTX, and SPARTA+.

However the agreement with experiment can be significantly improved by refining the protein structures using an energy function that includes a force field and a solvation term (OPLS-AA/L with the GB/SA continuum solvent model) and a chemical shift term in the program PHAISTOS. This refinement also results in structures with predicted trans-hydrogen bond coupling constants ($^{h3}J_{NC'}$) in good agreement with experiment indicating that the refined protein structures reflect the structures in solution. Comparison of average hydrogen bond geometries to those of high-resolution (<1.35 Å) X-ray structures reveals that the structural refinement improves the predicted δ_H-values through relatively small changes in the hydrogen bond geometry distribution.

Structural refinement without chemical shifts (i.e. using only the OPLS-AA/L + Generalized Born solvation energy) or combined with CamShift has relatively little effect on the predicted δ_H-values, while the predicted $^{h3}J_{NC'}$ values are in slightly worse agreement with experiment compared to using X-ray structures or ProCS-refined structures. This is not surprising given the fact that CamShift and similar empirical methods were designed to be insensitive to relatively small changes in protein structure in order to offer robust chemical shift predictions based on X-ray structures of varying accuracy. Structural refinement based on other empirical shift predictors, such as SHIFTS, SHIFTX, and SPARTA+, were not tested mainly because an efficient interface to PHAISTOS requires a complete re-implementation of the method. However, based on our comparison to the QM-calculations (Table 1 and Fig. 1) we do not think the conclusions will be substantially different. Our data, and that of Vila *et al.* [14], suggests that QM-derived chemical shift predictors are sufficiently accurate to extract small changes in structure and dynamics from experimentally measured protein chemical shifts.

We are currently working on implementing a QM-based chemical shift prediction method for the remaining H, C, and N nuclei in a protein in ProCS (unfortunately, the source code of the CheShift method developed by Vila *et al.* for QM-based C chemical shift prediction is not available). The resulting ProCS/PHAISTOS interface should provide a powerful tool for chemical shift-based protein structure refinement.

The ensembles resulting from the simulations can be downloaded from DOI: http://dx.doi.org/10.5879/BILS/p000001

Implementations of ProCS and CamShift can be downloaded as separate modules for PHAISTOS under the terms of the GNU General Public License v3 from: http://github.com/jensengroup/

Supporting Information

Supporting Information S1 Section S1: Time evolution of energies and chemical shift RMSDs during MCMC simulation. Figures S1–S3: Details of Monte Carlo energies and chemical shift RMSDs over time for the presented simulations. **Section S2: Parametrization of chemical shift contributions due to hydrogen bonding interactions to carboxylic acids and alcohols.** Figure S4: Sketches showing the geometric parameters and the systems used in the modeling of chemical shift contributions due to hydrogen bonding. **Section S3: Model for solvent exposed amide protons.** Table S1: Chemical shift contributions due to hydrogen bonding to water molecules. Figure S5: Local minima of NMA-water dimer.
(PDF)

Author Contributions

Conceived and designed the experiments: ASC KLL TH JHJ. Performed the experiments: ASC TEL MB WB. Analyzed the data: ASC TEL JHJ. Wrote the paper: ASC JHJ.

References

1. Mulder FAA, Filatov M (2010) Ab initio NMR chemical shift data and shielding calculations: Emerging tools for protein structure determination. Chem Soc Rev 395: 578–590.
2. Moon S, Case DA (2001) A new model for chemical shifts of amide hydrogens in proteins. J Biomol NMR 38: 139–150.
3. Xu XP, Case DA (2001) Automated prediction of 15N, $13C^{\alpha}$, $13C^{\beta}$ and 13C chemical shifts in proteins using a density functional database. J Biomol NMR 21: 321–333.
4. Shen Y, Bax A (2007) Protein backbone chemical shifts predicted from searching a database for torsion angle and sequence homology. J Biomol NMR 38: 289–302.
5. Neal S, Nip AM, Zhang H, Wishart DS (2003) Rapid and accurate calculation of protein 1h and 13C and 15N chemical shifts. J Biomol NMR 26: 215–240.
6. Meiler J (2003) PROSHIFT: Protein chemical shift prediction using artificial neural networks. J Biomol NMR 26: 25–37.
7. Kohlhoff KJ, Robustelli P, Cavalli A, Salvatella X, Vendruscolo M (2009) Fast and accurate pre-dictions of protein NMR chemical shifts from interatomic distances. J Am Chem Soc 131: 13894–13895.
8. Wishart D, Case DA (2001) Use of chemical shifts in macromolecular structure determination. Methods Enzymol 338: 3–34.
9. Case DA (2013) Chemical shifts in biomolecules. Curr Opin Struct Biol 23: 172–176.
10. Parker LL, Houk AR, Jensen JH (2006) Cooperative hydrogen bonding effects are key determinants of backbone amide proton chemical shifts in proteins. J Am Chem Soc 128: 9863–9872.
11. Barfield M (2002) Structural dependencies of interresidue scalar coupling ^{h3}J nc' and donor 1H chemical shifts in the hydrogen bonding regions of proteins. J Am Chem Soc 124: 4158–4168.
12. Cornilescu G, Ramirez BE, Frank MK, Clore MG, Gronenborn AM, et al. (1999) Correlation between ^{h3}J nc' and hydrogen bond length in proteins. J Am Chem Soc 121: 6275–6279.
13. Vila JA, Scheraga HA (2009) Assessing the accuracy of protein structures by quantum mechanical computations of 13C(alpha) chemical shifts. Acc Chem Res 42: 1545–1553.
14. Vila JA, Arnautova YA, Martin OA, Scheraga HA (2009) Quantum-mechanics-derived 13Ca chemical shift server (cheshift) for protein structure validation. Proc Natl Acad Sci 106: 16972–16977.
15. Boomsma W, Frellsen J, Harder T, Bottaro S, Johansson KE, et al. (2013) PHAISTOS: a framework for markov chain monte carlo simulation and inference of protein structure. J of Comp Chem 00: 000-000, DOI: 10.1002/jcc.23292.
16. Kaminski GA, Friesner RA (2001) Evaluation and reparametrization of the OPLS-AA force field for proteins via comparison with accurate quantum chemical calculations on peptides. J Phys Chem B 105: 6474–6487.
17. Qiu D, Shenkin PS, Hollinger FP, Still WC (1997) The GB/SA continuum model for solvation: A fast analytical method for the calculation of approximate born radii. J Phys Chem A 101: 3005–3014.
18. Czinki E, Császár AG (2004) On NMR isotropic chemical shift surfaces of peptide models. J Mol Struct (THEOCHEM) 675: 107–116.
19. Rablen PR, Pearlman SA, Finkbiner J (1999) A comparison of density functional methods for the estimation of proton chemical shifts with chemical accuracy. J Phys Chem A 103: 7357–7363.
20. Pople JA (1956) Proton magnetic resonance of hydrocarbons. J Chem Phys 24: 1111.
21. Pople JA (1958) Molecular orbital theory of aromatic ring currents. Mol Phys 1: 175–180.
22. Christensen AS, Sauer SPA, Jensen JH (2011) Definitive benchmark study of ring current effects on amide proton chemical shifts. J Chem Theory Comput 7: 2078–2084.
23. Jin L, Briggs SL, Chandrasekhar S, Chirgadze NY, Clawson DK, et al. (2000) Crystal structure of human parathyroid hormone 1-34 at 0.9 å resolution. J Biol Chem 275: 27238–27244.
24. Jain R, Bally T, Rablen PR (2009) Calculating accurate proton chemical shifts of organic molecules with density functional methods and modest basis sets. J Org Chem 74: 4017–4023.
25. Bottaro S, Boomsma W, Johansson KE, Andreetta C, Hamleryck TW, et al. (2011) Subtle monte carlo updates in dense molecular systems. J Chem Theory Comput 8: 695–702.
26. Dolinsky TJ, Nielsen JE, McCammon JA, Baker NA (2004) PDB2PQR: an automated pipeline for the setup, execution, and analysis of poisson-boltzmann electrostatics calculations. Nucl Acids Res 32: W665–W667.
27. Dolinsky TJ, Czodrowski P, Li H, Nielsen JE, Jensen JH, et al. (2007) PDB2PQR: expanding and upgrading automated preparation of biomolecular structures for molecular simulations. Nucl Acids Res 35: W522–W525.

28. Cordier F, Grzesiek S (1999) Direct observation of hydrogen bonds in proteins by interresidue 3hJNC' scalar couplings. J Am Chem Soc 121: 1601–1602.
29. Bax A (2003) Weak alignment offiers new nmr opportunities to study protein structure and dynamcs. Prot Sci 12: 1–16.
30. Rieping W, Habeck M, Nilges M (2005) Inferential structure determination. Science 308: 303–306.
31. Hamelryck T, Borg M, Paluszewski M, Paulsen J, Frellsen J, et al. (2010) Potentials of mean force for protein structure prediction vindicated, formalized and generalized. PLoS ONE 5: e13714.
32. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, et al. (2000) The protein data bank. Nucl Acids Res 28: 235–242.
33. Li H, Robertson AD, Jensen JH (2005) Very fast empirical prediction and rationalization of protein pKa values. Proteins 61: 704–721.
34. Zhang H, Neal S, Wishart D (2003) RefDB: a database of uniformly referenced protein chemical shifts. J Biomol NMR 25: 173–195.
35. Ulrich EL, Akutsu H, Doreleijers JF, Harano Y, Ioannidis YE, et al. (2008) Biomagresbank. Nucl Acids Res 36: 402–408.
36. Markwick PRL, Sprangers R, Sattler M (2003) Dynamic effects on j-couplings across hydrogen bonds in proteins. J Am Chem Soc 125: 644–645.
37. Frisch MJ, Trucks GW, Schlegel HB, Scuseria GE, Robb MA, et al. (2004) Gaussian 03, Revision C.02. Gaussian, Inc., Wallingford, CT.
38. Losonczi JA, Andrec M, Fischer MW, Prestegard JH (1999) Order matrix analysis of residual dipolar couplings using singular value decomposition. J Magn Reson 138: 334342.
39. Lindorff-Larsen K, Best RB, DePristo MA, Dobson CM, Vendruscolo M (2005) Simultaneous determination of protein structure and dynamics. Nature 433: 128–132.
40. Shen Y, Bax A (2010) SPARTA+: a modest improvement in empirical NMR chemical shift pre- diction by means of an artificial neural network. J Biomol NMR 48: 13–22.
41. Fenwick RB, Esteban-Martn S, Richter B, Lee D, Walter KFA, et al. (2011) Weak long-range correlated motions in a surface patch of ubiquitin involved in molecular recognition. J Am Chem Soc 133: 10336–10339.
42. Lange OF, Lakomek NA, Fars C, Schrder GF, Walter KFA, et al. (2008) Recognition dynamics up to microseconds revealed from an rdc-derived ubiquitin ensemble in solution. Science 320: 1471–1475.
43. Richter B, Gsponer J, Várnai P, Salvatella X, Vendruscolo M (2007) The mumo (minimal under-restraining minimal over-restraining) method for the determination of native state ensembles of proteins. J Biomol NMR 37: 117–135.
44. Lindorff-Larsen K, Best R, DePristo M, Dobson C, Vendruscolo M (2004) Simultaneous determination of protein structure and dynamics. Nature 433: 128–132.
45. Cornilescu G, Marquardt J, Ottiger M, Bax A (1998) Validation of protein structure from anisotropic carbonyl chemical shifts in a dilute liquid crystalline phase. J Am Chem Soc 120: 6836–6837.
46. Vijay-Kumar S, Bugg C, Cook W (1987) Structure of ubiquitin refined at 1.8 a resolution. J Mol Biol 194: 531–544.
47. Ramage R, Green J, Muir T, Ogunjobi O, Love S, et al. (1994) Synthetic, structural and biological studies of the ubiquitin system: the total chemical synthesis of ubiquitin. Biochem J 299: 151–158.
48. Alexeev D, Barlow PN, Bury SM, Charrier JD, Cooper A, et al. (2003) Synthesis, structural and biological studies of ubiquitin mutants containing (2s, 4s)-5-uoroleucine residues strategically placed in the hydrophobic core. ChemBioChem 4: 894–896.

FragBuilder: An efficient Python library to setup quantum chemistry calculations on peptides models

Anders S. Christensen, Thomas Hamelryck, Jan H. Jensen (2014) FragBuilder: An efficient Python library to setup quantum chemistry calculations on peptides models. *PeerJ* 2:e277.

PeerJ

FragBuilder: an efficient Python library to setup quantum chemistry calculations on peptides models

Anders S. Christensen[1], Thomas Hamelryck[2] and Jan H. Jensen[1]

[1] Department of Chemistry, University of Copenhagen, Copenhagen, Denmark
[2] Department of Biology, University of Copenhagen, Copenhagen, Denmark

ABSTRACT

We present a powerful Python library to quickly and efficiently generate realistic peptide model structures. The library makes it possible to quickly set up quantum mechanical calculations on model peptide structures. It is possible to manually specify a specific conformation of the peptide. Additionally the library also offers sampling of backbone conformations and side chain rotamer conformations from continuous distributions. The generated peptides can then be geometry optimized by the MMFF94 molecular mechanics force field via convenient functions inside the library. Finally, it is possible to output the resulting structures directly to files in a variety of useful formats, such as XYZ or PDB formats, or directly as input files for a quantum chemistry program. FragBuilder is freely available at https://github.com/jensengroup/fragbuilder/ under the terms of the BSD open source license.

Subjects Biochemistry, Computational Biology, Computational Science
Keywords Peptides, Computational chemistry, Molecular modeling, Proteins, Biochemistry

INTRODUCTION

Modeling of chemical properties of proteins is a challenging task in modern computational biochemistry, mainly due to the large number of atoms that need to be treated computationally, compared to the computational speed of modern computers. Although theoretical methods to treat large systems are being developed, it is computationally more feasible to investigate properties of small, representative, protein-like structures, such as peptides. For example, calculations on peptides have been used to parametrize protein-specific molecular mechanics force fields, and models for NMR properties of proteins such as chemical shifts and spin-spin coupling constants (*Mackerell, 2004*; *Vila et al., 2009*; *Case, Scheurer & Brüschweiler, 2000*).

Recently, we have used the presented Python library to carry out calculations on peptides modeling the backbone of a protein in the parametrization of amide proton chemical shifts (*Christensen et al., 2013*). Since this study, we have carried out more than 1.5 million quantum mechanical geometry optimization and NMR shielding calculations on peptides in order to extend our model of protein chemical shifts. Naturally, an efficient and stable method is needed in order to generate such a number of peptide models.

Two recent programs that can generate peptide structures are the Ribosome program (*Srinivasan, 2013*) and the PeptideBuilder library (*Tien et al., 2013*). The Ribosome

Submitted 23 December 2013
Accepted 27 January 2014
Published 4 March 2014

Corresponding author
Anders S. Christensen,
andersx@nano.ku.dk

Academic editor
Tomas Perez-Acle

Additional Information and Declarations can be found on page 11

DOI **10.7717/peerj.277**

program is written in FORTRAN and thus difficult to extend and therefore not ideal for use in an automated, scripting fashion. The PeptideBuilder library is written in Python and is therefore very attractive for this purpose. Our library which is presented here is very similar to PeptideBuilder, but offers a number of additional features which we found necessary for our purpose. Most importantly, our library includes methods for geometry optimization with a molecular mechanics force field, efficient conformational sampling from continuous probability distributions and lastly output to a variety of output formats or, optionally, directly as input file for a quantum chemistry program. Currently Gaussian 09 (*Frisch et al., 2009*) is supported via specialized classes, and nearly 100 additional file formats are supported through the file writer.

METHODS

FragBuilder is implemented in Python and is a library that can be imported and used in simple Python scripting style. Python is attractive, since a very large number of scientific libraries are already available in Python, and thus easy to extend and combine with new code. FragBuilder is implemented using the Open Babel library as back-end for handling the molecular structure of the peptide via existing classes and methods (*O'Boyle et al., 2011*). The methods present in FragBuilder thus have access to a multitude of existing chemistry and cheminformatics related library routines which are maintained separately by Open Babel. Especially, the code for manipulating a molecular structure, molecular mechanics and file writers from Open Babel are used in FragBuilder. FragBuilder also comes with the BASILISK library which can sample protein backbone and side chain conformations from a joint probability distribution (*Harder et al., 2010*).

The only dependencies for running FragBuilder are the NumPy mathematics library (*Oliphant, 2006*) and Open Babel with Python bindings. These packages are already available through package managers on virtually every recent Linux distribution, or otherwise freely available and open source.

FUNCTIONALITY AND USAGE

The functionality to create a peptide is implemented in the `Peptide` class which is imported from the `fragbuilder` module. A typical work flow creates a peptide, defines torsion angles, performs a constrained geometry optimization and finally writes the resulting structure to a file. A chart describing a typical use case is displayed in Fig. 1, and detailed examples of the functionality of FragBuilder are given below.

Furthermore FragBuilder has classes to easily access the BASILISK library, read PDB files and write input files for Gaussian 09. An overview of the available class as well as a brief description of each can be found in Table 1.

Creating peptides

The structure of a peptide molecule is generated as a Python object by using the `Peptide` class instantiated with the sequence as argument. The `Peptide` class has access to classes for each type of residues which each contain a structure for that residue in XYZ format. Routines from Open Babel are then used to automatically rotate, translate, and connect

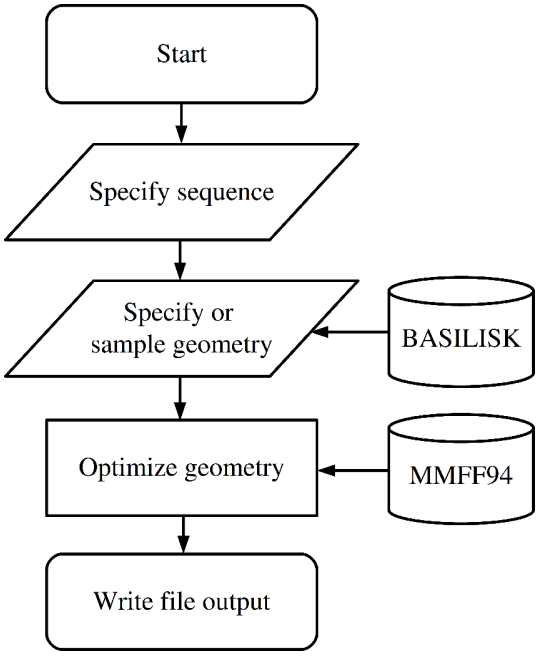


Figure 1 Flowchart describing the use of FragBuilder. Simple chart of a common workflow using FragBuilder. First a peptide is generated from the sequence. Then torsion angles are set — either specified manually or sampled through BASILISK and a quick geometry optimization is performed using the MMFF94 force field. Finally, the structure is written to a file.

Table 1 Overview of classes included in the FragBuilder library.

Class name	Description
`Peptide`	Class to create and manipulate a peptide structure and write output files
`Basilisk_DBN`	Wrapper class for direct access to the BASILISK library
`PDB`	Class to extract angles, sequence, etc. from a PDB file
`G09_opt`, `G09_NMR`, `G09_energy`	Classes to create input files for QM calculations in Gaussian 09

the residues. Finally the structure is stored in the `Peptide.molecule` class variable as an Open Babel `OBMol` object.

The sequence interpreted uses the single letter abbreviation for each amino acid. E.g., `Peptide("GLG")` will create a glycine–leucine–glycine tripeptide molecule which can then be manipulated through the interface. The minimal code to achieve this could be:

```
1 from fragbuilder import Peptide
2 pep = Peptide("GLG")
```

As default values, the ϕ, ψ and ω backbone torsion angles are set to $-120°, 140°$ and $-180°$, which corresponds to a typical extended β-strand. The side chain torsion χ angles are set so two neighboring side chains will not have steric clashes when no side chain torsion angle input is given. After the peptide has been instantiated, the structure can

Table 2 Overview of the basic methods in the `Peptide` class. See the text for detailed descriptions of each method.

Method name	Description
`set_bb_angles`	Set the backbone ϕ/ψ-angles for a residue
`set_chi_angles`	Set the side chain χ-angles for a residue
`get_bb_angles`	Read the backbone ϕ/ψ-angles for a residue
`get_chi_angles`	Read the side chain χ-angles for a residue
`sample_bb_angles`	Sample the backbone ϕ/ψ-angles for a residue using the BASILISK library
`sample_chi_angles`	Sample the side chain χ-angles for a residue using the BASILISK library
`optimize`	Perform a molecular mechanics optimization using the MMFF94 force field
`regularize`	Perform the regularization procedure to remove steric clashes
`write_pdb`	Write the peptide structure to a PDB file
`write_xyz`	Write the peptide structure to an XYZ file
`write_file`	Write the peptide structure to one of the nearly 100 file types supported by Open Babel

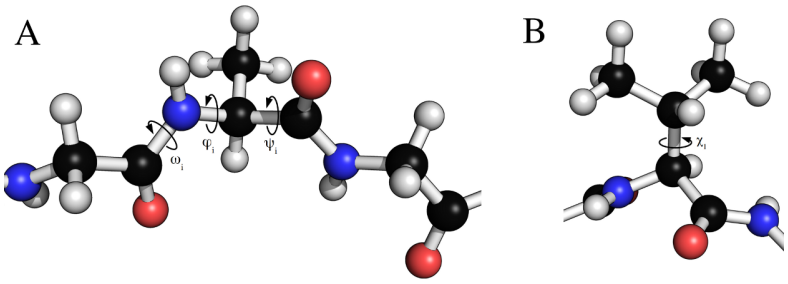


Figure 2 Torsion angles that can be treated by FragBuilder. Examples of dihedral angles that can be set via FragBuilder. In (A) the backbone ω, ϕ and ψ torsion angles are shown for the i'th alanine residue of a peptide strand. In (B), the χ_1 torsion angle is shown for a valine side chain.

be manipulated through built-in methods. Several convenient methods of the `Peptide` class are presented in the next sections. An overview of some of the basic methods of the `Peptide` class can be seen in Table 2.

Setting dihedral angles

The `Peptide` class allows for dihedral angles to be manually specified through setter and getter type functions that set or read backbone and side chain torsion angles. Examples of torsion angles that can be set in FragBuilder are shown in Fig. 2.

For example, making a glycine–leucine–glycine peptide and setting the backbone angles to $\phi = -60.0°$ and $\psi = -30.0°$, and side chain angles to $\chi_1 = 180°$ and $\chi_2 = 60°$ of the leucine (residue 2) can be done through the following code:

```
pep = Peptide("GLG")
pep.set_bb_angles(2, [-60.0, -30.0])
pep.set_chi_angles(2, [180.0, 60.0])
```

This way it is possible to precisely specify dihedral angles manually. This code can be used, for instance, to set up a scan of torsion angles or making peptides with geometries extracted from experimental structures. An example of a scan is shown in Fig. 3. This scan was created in the following manner:

```
pep = Peptide("GLG")
for i in range(10):
    pep.set_bb_angles(2, [-120.0, 100.0+20.0*i])
    pep.write_xyz("pep_%i.xyz" % (i))
```

The method `Peptide.write_xyz()` writes the structure to a file in XYZ format and is described later in this section.

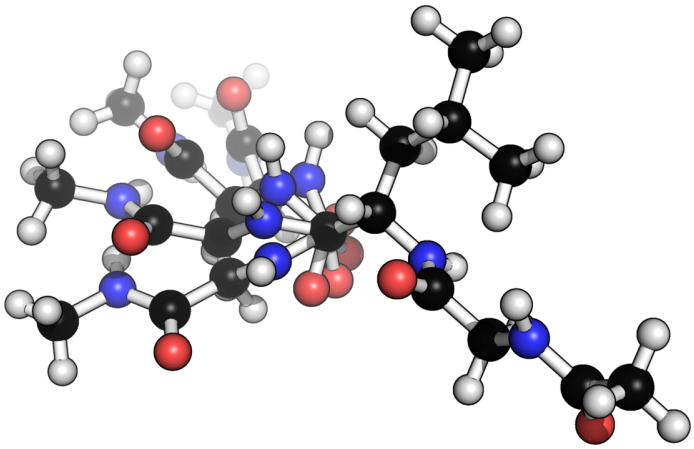


Figure 3 Example of four different conformers of a glycine–alanine–glycine tri-peptide. Generated from a scan over the ψ backbone torsion angle of the alanine residue.

Sampling dihedral angles from BASILISK

In addition to manual specification of torsion angle values, it is possible to set these to values from predefined distributions, such as the Ramachandran-plot for backbone angles or rotamer distributions for side chain angles. This allows for fast and efficient sampling of realistic peptide conformations and rotamer distribution without the need for a molecular dynamics or Monte Carlo simulation. For this purpose FragBuilder includes the BASILISK library and convenient methods to access BASILISK from the `Peptide` class.

BASILISK is a dynamic Bayesian network trained on a large set of representative structures from the Protein Data Bank (*Berman et al., 2000*) and is able to sample backbone angles and side chain angles. BASILISK makes use of directional statistics — the statistics of angles, orientations and directions — to formulate a well-defined joint probability distribution over side and main chain angles. Backbone angles are essentially sampled

from the Ramachandran-plot via BASILISK. Similarly, side chain angles are sampled from corresponding rotamer distributions. The distributions offered by the BASILISK library are continuous, in contrast to most approaches based on discrete rotamer libraries. BASILISK can sample side chain angles either in a backbone conformation-dependent mode or -independent mode (where backbone dependency is the default behavior). The random seed can be set explicitly via the `fragbuilder.set_seed()` function. If no seed is supplied the seeding will be random.

The methods `Peptide.sample_bb_angles()` and `Peptide.sample_bb_angles()` allows the user to simultaneously sample and set the torsion angles of a residue. The methods return the new sets of sampled angles so they are known to the user directly.

The following code will create a glycine–leucine–glycine peptide and set the backbone and side chain angles of the second residue (leucine) to values that are sampled from BASILISK. The values of the sampled angles are stored in the `new_bb` and `new_chi` variables.

```
from fragbuilder import Peptide, set_seed
set_seed(42)
pep = Peptide("GLG")
new_bb = pep.sample_bb_angles(2)
new_chi = pep.sample_chi_angles(2)
```

It is also possible to get samples from BASILISK via FragBuilder by using the `fragbuilder.Basilisk_DBN` class which provides direct access to the sampler in the BASILISK library. This class is used to obtain samples of ϕ/ψ angles from the Ramachandran-plot or sets of χ angles from rotamer distribution without first creating a peptide.

For instance, a random set of χ angles (`chi`), ϕ/ψ angles (`bb`), and their corresponding log-likelihood (`ll`) in the probability distribution can be obtained as follows (here for a Leucine ("L") residue):

```
from fragbuilder import Basilisk_DBN
dbn = Basilisk_DBN()
# Amino acid type as argument
chi, bb, ll = dbn.get_sample("L")
```

10,000 of such samples from the above code was used to create the Ramachandran plot and rotamer distribution of leucine which can be seen in Figs. 5A and 5B, respectively.

Capping peptides

One aspect of carrying out quantum mechanical calculations on peptide fragments is the way the peptide strands are terminated or capped. This can be important, since the properties calculated from a quantum mechanical calculation may be affected by how the protein is truncated to a model peptide. The specific type of cap is controlled by setting the keywords `nterm` and `cterm` keywords (for the N-terminus and C-terminus, respectively) when the peptide object is created.

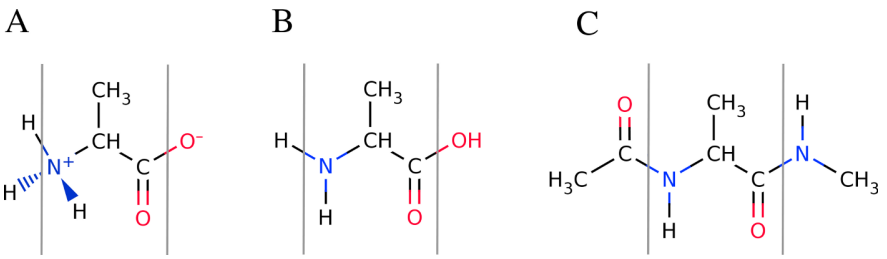


Figure 4 Overview of the available peptide-capping schemes available in FragBuilder. All three examples show an alanine residue (shown between a set of gray lines). In (A), the caps are the N- and C-termini in their charged states. In (B) the caps are the N- and C-termini in their neutral states. In (C) the caps are methyl groups. Caps can be mixed and matched according to the user's specifications.

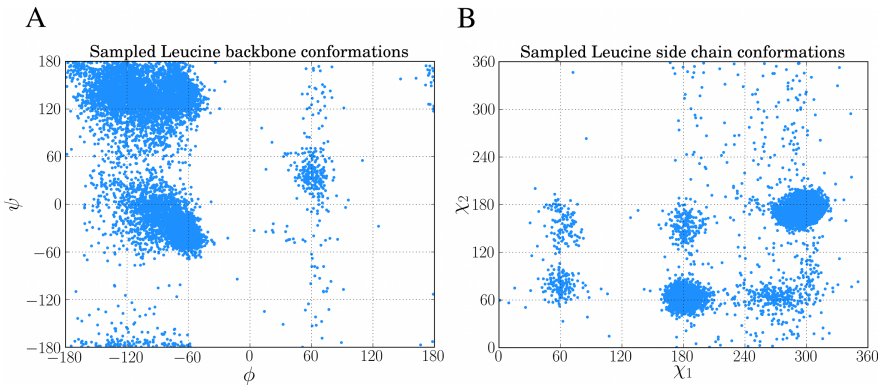


Figure 5 Examples of sampling dihedral angles through BASILISK in FragBuilder. 10,000 samples from BASILISK are shown for a leucine residue. (ϕ, ψ) backbone torsion angle pairs are shown in (A) and (χ_1, χ_2) side chain torsion angle pairs are shown in (B).

By default, FragBuilder generates methyl caps by adding a CH_3–C(=O)- group to the N-terminus and an -NH–CH_3 group to the C-terminus (i.e., if the keywords are not set). This corresponds to setting both keywords (`nterm` and `cterm`) to `"methyl"`. Additionally, it is possible to cap the ends of the peptide as normal N- and C-termini (amine or carboxyl groups, respectively) which can be set to either a charged or a neutral state. A charged or neutral terminus is specified by passing the values `"charged"` or `"neutral"`, respectively. See Fig. 4 for a schematic of the three possible types of caps.

For instance, a glycine–leucine–glycine residue with a positively charged N-terminus and a neutral C-terminus is generated by the following code:

```
pep = Peptide("GLG", nterm="charged", cterm="neutral")
```

Optimization

When generating peptides with a specific set of dihedral angles the structure may, in some cases, contain steric clashes. We found this prevented us from starting quantum mechanical geometry optimization on the structures, even when these were generated to match angles from experimental structures. Typical problems with these structures were

SCF convergence issues and very large molecular gradients which cause the program to fail. In some cases, problems with large molecular gradients may be alleviated by adjusting the step-size in the optimizer, but this must be investigated on a case-to-case basis. It is therefore advantageous to remove steric clashes before any quantum mechanical calculation is carried out.

For the reasons mentioned above, FragBuilder offers specialized molecular mechanics optimization routines, specifically designed to constrain the dihedral angles of peptides while removing steric clashes. Optimization is performed through Open Babel which provides access to several force fields and a number of optimizers. The MMFF94 force field (*Halgren, 1996*) is arguably the most advanced force field for biomolecules in Open Babel and is used exclusively in FragBuilder along with the conjugate gradient method. FragBuilder offers three kinds of optimization methods in the `Peptide` class.

The method `Peptide.optimize()` will perform a conjugate gradient optimization of the peptide with no restraints, until the default convergence criterion of Open Babel is reached ($\Delta E < 1.0 \times 10^{-6}$ kcal/mol or a max of 500 steps). Another option is to impose harmonic constraints on all dihedral angles. This is achieved through an extra keyword, i.e., `Peptide.optimize(constraint=True)`. This will perform a conjugate gradient minimization through Open Babel with harmonic potentials on ϕ, ψ and ω backbone angles as well as all side chain χ angles.

A harmonic potential does not keep torsion angles completely fixed during optimization, and after convergence they deviate slightly from the starting values. For situations where this is problematic, FragBuilder is offering a routine termed "regularizing" which is accessed via the `Peptide.regularize()` method.

Regularizing cycles between a few constrained geometry optimization steps and resetting the dihedral angles to the initially specified angles, until self consistency is reached. A default regularization cycles 10 times between 50 conjugate gradient steps and angle resets. In most cases this converges the constrained optimization to less than 0.002° from the specified dihedral angles, which are then set to the specified values.

We found our regularization procedure with flexible bond lengths and angles through the MMFF94 force field to allowing convergence of QM calculations in many cases, which would have been hindered by steric clashes due to fixed bond length and angles.

A similar approach to avoid spurious conformations has been adopted by Vila et al. in the creation of the CheShift chemical shifts predictor, which is parametrized from quantum mechanical calculations on model peptides (*2009*). Here bond angles and lengths are simply set to the standard values of the ECEPP/3 force field (*Nemethy et al., 1992*). Subsequently the internal energy of the peptide is calculated with the ECEPP-05 force field and any conformation with an internal energy >30 kcal/mol is rejected as being unphysical.

Figure 6 shows an example of a tryptophan–aspartate–glycine peptide with methyl caps in which the backbone torsion angles are taken from the experimental structure of xylanase (PDB-code: 1XNB), residues 99–101. This choice of angles causes a clash between a hydrogen bonding O...H pair, and a geometry optimization at the B3LYP/6-31+G(d,p) level in Gaussian 09 could not start (at default settings) due to an excessively large

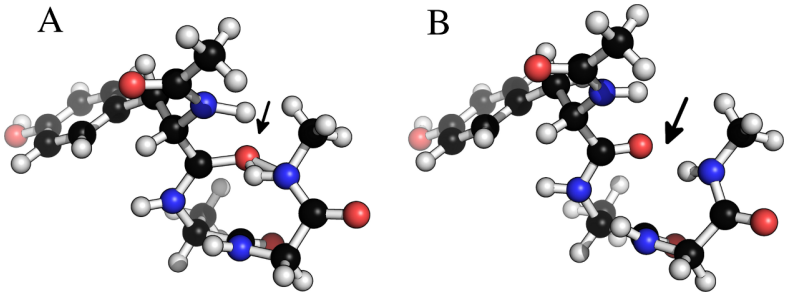


Figure 6 Removing clashes by regularization in a tryptophan–apartate–glycine peptide. In (A) the peptide clashes between the amide proton on the C-terminal methyl cap and the amide oxygen in residue 1. In (B) this clash has been removed by constrained relaxation during the regularization procedure. Both structures have identical ϕ, ψ and ω backbone torsion angles.

molecular gradient in the initial geometry. Regularization removes the clash, while retaining the specified dihedral angles, and allows the optimization to proceed.

A peptide can be created and regularized using the following code, which also prints the MMFF94 force field energy in units of kcal/mol:

```
pep = Peptide(sequence)
# The user can manipulate the structure here

pep.regularize()
print pep.get_energy()
```

Reading PDB files

While sampling and conformational scanning, etc., are efficient ways to generate new peptide conformations, it can be necessary to extract information about the conformation of a specific protein structure, usually given in PDB format. FragBuilder implements functionality to extract information about the amino acid sequence and dihedral angles from a structure in a PDB formatted file, which can then be stored or passed on in the program, for instance to methods in the `Peptide` class. This is carried out via the `fragbuilder.PDB` class which creates an object from a PDB file and offers methods to read the relevant information.

The following code example illustrates the basic usage of the `fragbuilder.PDB` module, and will print the amino acid type and dihedral angles of residue number 10 in the PDB file `"structure.pdb"`:

```
from fragbuilder import PDB

pdbfile = PDB("structure.pdb")
i = 10 # Residue number 10 in this example
print pdbfile.get_resname(i)
print pdbfile.get_bb_angles(i)
print pdbfile.get_chi_angles(i)
```

File output and interface to QM programs

Open Babel provides very flexible file readers and writers. The `Peptide` class wraps Open Babel with functions to directly write the geometry of a `Peptide` object to a file in XYZ or PDB format. This can be done simply as:

```
pep = Peptide(sequence)
pep.write_xyz("pep.xyz")
pep.write_pdb("pep.pdb")
```

It is also possible to write to any of the nearly 100 formats supported in Open Babel by using the method `Peptide.write_file(filetype, filename)` which offers direct access to Open Babel's `OBConversion.WriteFile()` method. For instance, an input file for the quantum chemistry program GAMESS (*Schmidt et al., 1993*) can be created with the following code:

```
pep.write_file("gamin", "pep.inp")
```

Here, the file type argument follows the Open Babel syntax, where `"gamin"` corresponds to the GAMESS input file format.

FragBuilder additionally offers an interface to write input-files for Gaussian 09, beyond the capabilities of Open Babel. Currently, it is possible to set up geometry optimization, single-point energy calculations and calculation of NMR shielding. An example for a simple workflow that will generate a file for geometry optimization of a peptide in Gaussian 09 at the B3LYP/6-31G(d) level (using the `fragbulder.G09_opt` class) is as follows:

```
from fragbuilder import Peptide, G09_opt

pep = Peptide(sequence)
# The user can manipulate the structure here

opt = G09_opt(pep)
opt.set_method("B3LYP/6-31G(d)")
opt.write_com("pep.com")
```

If no method or basis set is specified, the file writer defaults to PM6 (*Stewart, 2007*) for geometry optimization. Other classes that interface to Gaussian 09 are the `fragbuilder.G09_NMR` and `fragbuilder.G09_energy` classes, which are imported and instantiated similarly.

CONCLUSION

We have implemented routines to generate peptide models, from either specific geometries or efficient conformational sampling through the BASILISK library. We have furthermore implemented necessary code to perform constrained geometry optimizations of the peptide models, remove steric clashes and prepare the structure for use in a quantum

chemistry program. In addition, the file writers accommodate nearly 100 file formats, and are able to write input files for a number of chemistry programs through an interface to Open Babel.

The `Peptide` class wraps functionality from Open Babel offered through its Python interface. The molecular structure is stored as an Open Babel `openbabel.OBMol` object in the `Peptide.molecule` class variable. This means that developers and users effectively have access to all the tools present in Open Babel to further manipulate the structure, or extend FragBuilder by wrapping and combining functionality from Open Babel.

FragBuilder is open source and published under the BSD 2-Clause license. Note that the packaged BASILISK library is published under the GNU General Public License version 3. FragBuilder is freely available at https://github.com/jensengroup/fragbuilder/ where additional examples and full documentation can be found.

ACKNOWLEDGEMENTS

The authors would like to thank Casper Steinmann for valuable input during development of FragBuilder.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

Anders S. Christensen is funded by the Novo Nordisk STAR program. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
Novo Nordisk STAR program.

Competing Interests

The authors declare there are no competing interests.

Author Contributions

- Anders S. Christensen conceived and designed the experiments, performed the experiments, analyzed the data, contributed reagents/materials/analysis tools, wrote the paper, prepared figures and/or tables, reviewed drafts of the paper.
- Thomas Hamelryck contributed reagents/materials/analysis tools, wrote the paper, reviewed drafts of the paper.
- Jan H. Jensen conceived and designed the experiments, analyzed the data, wrote the paper, reviewed drafts of the paper.

Data Deposition

The following information was supplied regarding the deposition of related data:
https://github.com/jensengroup/fragbuilder/.

REFERENCES

Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE. 2000. The protein data bank. *Nucleic Acids Research* **28**:235–242 DOI 10.1093/nar/28.1.235.

Case DA, Scheurer C, Brüschweiler R. 2000. Static and dynamic effects on vicinal scalar *J* couplings in proteins and peptides: a MD/DFT study. *Journal of the American Chemical Society* **122**:10390–10397 DOI 10.1021/ja001798p.

Christensen AS, Linnet TE, Borg M, Boomsma W, Lindorff-Larsen K, Hamelryck T, Jensen JH. 2013. Protein structure validation and refinement using amide proton chemical shifts derived from quantum mechanics. *PLoS ONE* **8**:e84123 DOI 10.1371/journal.pone.0084123.

Frisch MJ, Trucks GW, Schlegel HB, Scuseria GE, Robb MA, Cheeseman JR, Scalmani G, Barone V, Mennucci B, Petersson GA, Nakatsuji H, Caricato M, Li X, Hratchian HP, Izmaylov AF, Bloino J, Zheng G, Sonnenberg JL, Hada M, Ehara M, Toyota K, Fukuda R, Hasegawa J, Ishida M, Nakajima T, Honda Y, Kitao O, Nakai H, Vreven T, Montgomery JA Jr, Peralta JE, Ogliaro F, Bearpark M, Heyd JJ, Brothers E, Kudin KN, Staroverov VN, Kobayashi R, Normand J, Raghavachari K, Rendell A, Burant JC, Iyengar SS, Tomasi J, Cossi M, Rega N, Millam JM, Klene M, Knox JE, Cross JB, Bakken V, Adamo C, Jaramillo J, Gomperts R, Stratmann RE, Yazyev O, Austin AJ, Cammi R, Pomelli C, Ochterski JW, Martin RL, Morokuma K, Zakrzewski VG, Voth GA, Salvador P, Dannenberg JJ, Dapprich S, Daniels AD, Farkas O, Foresman JB, Ortiz JV, Cioslowski J, Fox DJ. 2009. *Gaussian 09 Revision D.01*. Wallingford, CT: Gaussian Inc.

Halgren TA. 1996. Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. *Journal of Computational Chemistry* **17**:490–519 DOI 10.1002/(SICI)1096-987X(199604)17:5/6<490::AID-JCC1>3.0.CO;2-P.

Harder T, Boomsma W, Paluszewski M, Frellesen J, Johansson KE, Hamelryck T. 2010. Beyond rotamers: a generative, probabilistic model of side chains in proteins. *BMC Bioinformatics* **11**:306–318 DOI 10.1186/1471-2105-11-306.

Mackerell AD. 2004. Empirical force fields for biological macromolecules: overview and issues. *Journal of Computational Chemistry* **25**:1584–1604 DOI 10.1002/jcc.20082.

Nemethy G, Gibson KD, Palmer KA, Yoon CN, Paterlini G, Zagari A, Rumsey S, Scheraga HA. 1992. Energy parameters in polypeptides. 10. Improved geometrical parameters and nonbonded interactions for use in the ECEPP/3 algorithm, with application to proline-containing peptides. *The Journal of Physical Chemistry* **96**:6472–6484 DOI 10.1021/j100194a068.

O'Boyle NM, Banck M, James CA, Morley C, Vandermeersch T, Hutchinson GR. 2011. Open Babel: an open chemical toolbox. *Journal of Cheminformatics* **3**:33–46 DOI 10.1186/1758-2946-3-33.

Oliphant T. 2006. NumPy. *Available at http://www.numpy.org/* (accessed 10 December 2013).

Schmidt MW, Baldridge KK, Boatz JA, Elbert ST, Gordon MS, Jensen JH, Koseki S, Matsunaga N, Nguyen KA, Su SJ, Windus TL, Dupuis M, Montgomery JA. 1993. General atomic and molecular electronic structure system. *Journal of Computational Chemistry* **14**:1347–1363 DOI 10.1002/jcc.540141112.

Srinivasan R. 2013. Ribosome — program to build coordinates for peptides from sequence. *Available at http://folding.chemistry.msstate.edu/~raj/Manuals/ribosome.html* (accessed 10 December 2013).

Stewart J. 2007. Optimization of parameters for semiempirical methods V: modification of NDDO approximations and application to 70 elements. *Journal of Molecular Modeling* **13**:1173–1213 DOI 10.1007/s00894-007-0233-4.

Tien MZ, Sydykova DK, Meyer AG, Wilke CO. 2013. PeptideBuilder: a simple Python library to generate model peptides. *PeerJ* **1**:e80 DOI 10.7717/peerj.80.

Vila JA, Arnautova YA, Martin OA, Scheraga HA. 2009. Quantum-mechanics-derived $^{13}C^{\alpha}$ chemical shift server (*Che*Shift) for protein structure validation. *Proceedings of the National Academy of Sciences of the United States of America* **106**:16972–16977 DOI 10.1073/pnas.0908833106.