The success of many protein structure prediction methods critically depends on the existence of structural templates and our ability to detect them. It seems that known protein structures have reached a sufficient level of diversity for template-based structure prediction. In conserved regions, also a dynamic approach should be as accurate as static libraries. But in non-conserved regions, the better coverage of dynamic fragment search comes at the price of reduced precision. Another limitation is that fragments typically come in fixed size, with 9mer fragments being a popular choice (Bystroff et al., 1996; Holmes and Tsai, 2004; Simons et al., 1997). This is more for technical convenience rather than having a specific biological meaning. The actual instances of a structural motif often show variable lengths with extensions or deletions at the termini. I-Sites tries to address this problem by defining a set of overlapping, partially redundant motifs. The Rosetta fragment selection tool NMake searches a database of crystal structures with better than 2.5 Å resolution and pairwise sequence identity <50% (Rohl et al., 2004). All nine residue windows from the query sequence are compared to the database entries using a score that is based on sequence profile comparison and secondary structure match. The PSI-BLAST profiles of the query and the candidate fragments are compared using the city-block distance metric. This method has been extended by the FRazor dynamic fragment selection tool (Li et al., 2008). Using integer linear programming, FRazor combines the sequence profile similarity score with additional structural features (secondary structure, solvent accessibility and contact capacity) in order to improve the fragment selection. An alternative to fragment selection is the recent development of fragment sampling from probabilistic...
HHfrag: HMM-based fragment detection using HHpred

We use the standard HHpred toolchain (Hildebrand et al., 2009) to build a profile-HMM of the query sequence and the sequences in the template database. The fragment extraction routine uses the HHsearch algorithm for HMM-HMM comparison probes each fragment for recurrence in the structure database. Second, an ordered fragment map is compiled by finding the locally optimal regions of similarity between the query HMM segments and the HMMs in the structure database.

2 METHODS

Given a query sequence of unknown structure, our method builds a dynamic library by excision of fragments from a non-redundant structure database. First, the query HMM is divided into a set of overlapping HMM fragments of variable length (6–21 residues). The optimal boundaries of each query HMM segment are determined dynamically: HMM-HMM comparison probes each fragment for recurrence in the structure database. Second, an ordered fragment map is compiled by finding the locally optimal regions of similarity between the query HMM segments and the HMMs in the structure database.

2.1 Generation of HMM profiles

We use the standard HHpred toolchain (Hildebrand et al., 2009) to build a profile-HMM of the query sequence and the sequences in the template database. This step involves generation of multiple alignments with several rounds of PSI-BLAST (Altschul et al., 1997). In addition to a pure sequence-based score, predicted and observed secondary structure is taken into account. We use DSSP (Kabsch and Sander, 1983) to calculate eight-state secondary structure assignments for known structures from the template database. The query sequence is also converted to a profile HMM with predicted three-state secondary structure using PSIPRED (Jones, 1999). Each final

![Flowchart of the fragment extraction procedure](image)

Fig. 1. Flowchart of the fragment extraction procedure. (a) Procedure to determine fragment boundaries dynamically. (b) Fragment search using the dynamically extracted query fragments.

HMM comprises amino acid emission probabilities and secondary structure propensities and is built using HHmake with pseudo-count correction.

2.2 Template database

To strip down homologues from the PDB, we use the April 2010 build of PDBSelect25 (Griep and Hobohm, 2010) as template database. There are 4824 chains in the set sharing sequence identity up to 25%. For each structure, a profile HMM (sequence profile and secondary structure) is generated using the above procedure and the 3D coordinates are extracted. We refer to the resulting set of HMM/3D structure pairs as PDBSelect25-HMM template database.

2.3 Fragment extraction and assignment

The fragment extraction routine uses the HHsearch algorithm for HMM-HMM alignment (Söding, 2005). HHsearch scores via co-emission probability of the amino acid distributions at each profile column and through secondary structure match. In all HHsearch runs, the hit list includes all matches with probability ≥0.2. If there are no or less than 10 hits with probability greater than this threshold, additional hits with lower probability, if available, will be included until the number of hits is at least 10. Figure 1 shows a flowchart of our dynamic fragment search (a detailed description of HHfrag can be found in the Supplementary Material). The search consists of two phases: (i) identification of flexible query segments and (ii) fragment search. The first step is detailed in the next subsection. In the second phase, each query fragment HMM is compared against the template database using local HMM-HMM alignment (HHsearch with default parameters).
As an alternative to NNmake, we used the TorusDBN model by Boomsma et al.

All Rosetta 9mer and 3mer fragment libraries were generated using NNmake. A fragment is considered a true hit or true positive if the Cα RMSD to the native structure is below a threshold value (typically 1.5 Å). Since the RMSD is strongly length dependent, we also ran tests with a length-independent definition of true positives. However, we found that the overall picture of our results does not change (for details see Supplementary Material). We therefore use the conceptually simpler definition of a true positive based on RMSD (Kolodny et al., 2002). The precision is the percentage of true positives among the fragments assigned to the query. We report local residue-wise precision that assesses the percentage of true hits covering a specific residue. We also report the global precision measuring the percentage of true positives assigned to the entire target. The coverage is the percentage of target residues that are covered by at least one truly positive fragment (Li et al., 2008). Precision and coverage depend on the true positive threshold and increase if the threshold increases. Figures reporting all criteria are found in the Supplementary Material.

### 2.6 Fragment selection with Rosetta NNmake

All Rosetta 9mer and 3mer fragment libraries were generated using NNmake and the default Rosetta Fragments database taken from Rosetta v3.1. We used only PSIPRED secondary structure predictions and therefore the fragment generation program was started with the relevant command line arguments to suppress all other secondary structure prediction options.

### 2.7 Fragment sampling with TorusDBN

As an alternative to NNmake, we used the TorusDBN model by Boomsma et al. (2008) to sample protein conformations with a generative probabilistic model. For a given query sequence, we instantiated a TorusDBN model based on the sequence and predicted secondary structure and drew 100 random conformations. For calculation of the precision and accuracy, we used 9mer fragment structures excised from the full-length samples.

### 2.8 Decoy generation

To test our fragments in ab initio structure prediction, we have built a modified version of the standard Rosetta AbinitioRelax application from the Rosetta 3.1 C++ source distribution. We store the fragment library in Rosetta fragment format and directly feed it into AbinitioRelax in place of the standard 9mer library. We generate 1000 decoys per target. Each decoy is superimposed to the native structure of the target using local RMSD fitting and ranked by Cα RMSD and TM-score (Zhang and Skolnick, 2004).

### 3 RESULTS AND DISCUSSION

#### 3.1 From static to dynamic fragment libraries

Our initial goal was to create a static dictionary of fragments from profile HMMs. One of the issues that we wanted to address was the restriction to a fixed fragment length. To determine recurrent fragments of variable length, we use the approach outlined in Section 2.4. This approach redisCOVERs most of the motifs present in I-Sites but does not improve the coverage significantly, which makes sense intuitively because any attempt to describe a large cluster of fragments by a single profile will eventually decrease the sensitivity of the profile. We also find that many of the long fragments in our static library correspond to more than one I-Sites motif, suggesting that some fragments could be decomposed further into submotifs. Some proteins in the template database contain instances of the ‘full’ motif, while others have only ‘partial’ or even ‘minimal’ matches. This modularity is also observed in the I-Sites library itself (Fig. 2). Some I-Sites motifs are highly related in terms of sequence and structure and refer to a common core motif. We also found instances...
HHfrag: HMM-based fragment detection using HHpred

Comparison of the performance in fragment assignment between I-Sites and a classical dynamic approach (Rosetta fragments) shows that I-Sites as well as our own static library is highly specific, but the sequence coverage is insufficient for structure prediction (Fig. 3). On the other hand, Rosetta’s fragment extraction module achieves excellent coverage but tends to bury the good fragments in a vast number of low-quality hits.

3.2 Dynamic fragment detection

Based on the above observations, we decided to keep our fragments variable in length but make the fragment extraction routine truly dynamic with the aim of increasing the coverage while maintaining the current high level of precision. Figure 3 shows an example where our method may outperformed by Rosetta in terms of coverage. Rosetta reaches 87% coverage at an RMSD threshold of 1.4 Å, whereas HHfrag’s coverage is 76% at the same threshold. However, coverage alone is a misleading metric. Coverage counts the number of residues that are covered by at least one fragment with acceptable structure but does not assess how many of the assigned fragments are actually correct. The picture changes significantly if one considers also the precision of the fragments and asks oneself: if, at a given position in the query sequence, we pick one of the assigned fragments randomly, what is the chance that this fragment will have the correct structure? For the same target, HHfrag has 1.5-fold higher precision than Rosetta at the same RMSD cutoff.

The position-specific precision (Fig. 4; see also Supplementary Material) reveals that the quality of the assigned fragments is not uniform along the sequence of the query. Plots of the local precision usually demonstrate a characteristic shape. We observe well-expressed peaks of high-quality fragments connected by regions with no or very uncertain assignments. The high-quality regions exhibit a strong sequence signal and typically correspond to recurrent motifs similar to the I-Sites paradigms. The profiles of the I-Sites motifs always align to query regions where HHfrag assigns a high amount of truly positive fragments, 80±18% on average. The unassigned and/or low-quality regions show the highest variability and uncertainty, and need to be modeled using a brute-force approach during an ab initio structure prediction. The fact that HHfrag does not assign fragments to uncertain regions (‘white regions’ in the fragment map) should be considered a feature rather than a shortcoming, because it indicates that such areas require special treatment during modeling. However, current structure prediction protocols such as Rosetta AbinitioRelax may not be able to take advantage of this information (Supplementary Material).

The residue-wise precision diagrams have also similar patterns for both our and Rosetta’s fragment maps. The locations of the high-accuracy peaks along the target sequence are correlated. However, the Rosetta histograms display a peculiar triangular shape, whereas HHfrag assignments have a more block-like structure. The peaks in NNmake’s fragment map are sharp, and the precision quickly decreases as we move away from the maximum. This behavior can be explained by the lack of context variability of the Rosetta fragments and illustrates some of the disadvantages of using fragments of constant length.

The main advantage of HHfrag is therefore the ability to focus more precisely on the actual boundaries of the conserved regions. Gapped fragment assignments have a small impact on the global precision of the method provided that the database of structural templates is diverse enough (Section 3.3). However, the ability to detect insertions and deletions may have a decisive advantage if the database contains only few instances of a structural motif.
3.3 Benchmark

We used 105 target sequences from the CASP 9 competition (http://www.predictioncenter.org/casp9/) to test the performance of our dynamic fragment search method. The experimental structures of the benchmark proteins have been published after May 2010 and therefore do not appear in our template database. For each target, we built a position-specific fragment map (see Section 2) and compared its performance to a reference Rosetta 9mer fragment map. The $C_\alpha$ coordinates of all fragments were superimposed onto the native structure of the target based on the fragment map (Fig. 5 shows an example).

Figure 6 shows the distribution of fragment lengths found by dynamic HHfrag searches. The distribution of fragment lengths peaks at a value of seven, but shows significant probability for detecting longer fragments (the average length is $10.3 \pm 3.6$).

The average sequence coverage of HHfrag is $71 \pm 13\%$ (Fig. 7). If we restrict the analysis to residues in regular secondary structure, the coverage rises to $84 \pm 14\%$. The percentage of residues that remain completely unassigned (white regions) is $19 \pm 12\%$.

This is a significant improvement over static libraries such as I-Sites and an acceptable loss in coverage compared with Rosetta’s fragment selection module reaching $90 \pm 6\%$ coverage. Figure 7 provides also a summary of the global precision for all targets in the benchmark. On average, HHfrag obtains a precision of $62 \pm 16\%$, which is a significant improvement over Rosetta’s fragment selection with a precision of $38 \pm 17\%$. The improvement in the precision is two-fold on average and for some targets achieves a dramatic increase by a factor of 4 to 6. This improvement is clearly consistent across all three CASP target categories.

We also compared the quality of fragments extracted with HHfrag with generative statistical models that capture local sequence-structure correlations. We used the TorusDBN model of Boomsma et al. (2008) to draw 100 random configurations for each of the 105 CASP targets and compared the local quality of the sampled
This shows that coverage is not the only quantity that should be
detected with Rosetta NNmake, 4 targets remained for which Rosetta
picked the 11 shortest single-domain targets with lengths up to 150
residues to ensure that Rosetta has a good chance to predict their
conformations with the HHfrag fragment assignments (Fig. 7; see
Section 2 and Supplementary Material for more details). This
comparison shows that HHfrag produces more accurate and less
heterogeneous local structural fragments than the TorusDBN model
(average precision 28.4 ± 16.3%, average coverage 91.3 ± 6.5%). At
this instance, we would like to point out that comparison of fragment
selection and fragment sampling methods is a non-trivial issue. In
HHfrag, not all residues are assigned the same number of fragments,
because we use fragments of variable length and because HHsearch
assigns a variable number of hits to a query segment (on average
there are 39 such hits). This will potentially lower the coverage of
HHfrag’s fragment selection in comparison to methods that use a
fixed number of top scoring fragments such as NNmake. However,
coverage is not the only important parameter, and the major goal of
HHfrag is to find a trade-off between precision and coverage. There
is a fundamental difference here between fragment selection and
fragment sampling. In fragment selection by searching a database,
some residues (typically located in loops) will never be assigned
a fragment for various reasons: either these residues are highly
non-conserved and the fragment detection fails or the database of
templates is not comprehensive enough. In probabilistic fragment
sampling, on the other hand, there is always a non-zero change that
every residue will be covered as long as we sample long enough.
In the limit of infinitely many samples drawn from a generative
probabilistic model such as TorusDBN, the coverage will approach
one, but the precision may drop to very low values if the native
fragment is not contained in the high probability density region.
This shows that coverage is not the only quantity that should be
looked at. Also for practical purposes, we want a small number of
fragments, which is valid for HHfrag.

Of all true positive fragments (RMSD < 1.5 Å) found by HHfrag
90.8% have uninterrupted structure. However, gapped fragments
have been extracted at least once for 98 out of all 105 benchmark
proteins. These gaps typically represent very small insertions or
deletions in or around the central region of the motif (see the
Supplementary Material). With the current degree of structural
diversity of the PDBS25-HMM database, the gapped assignments
are not strongly influencing the overall performance of HHfrag.
However, 8.4% of the best-fitting fragments per query position are in
fact part of gapped assignments, which suggests that gap detection
may be useful when the number of available local templates is
limited.

3.4 Impact on decoys

The performance of our variable-length fragment libraries with
refined precision was tested in ab initio protein folding experiments
(Fig. 8; see also Supplementary Material). After modifying the
original Rosetta AbinitioRelax protocol to accept fragments of
variable length, we generated decoys for a subset of the proteins
in our benchmark. Initially, we found 15 targets for which the
AbinitioRelax protocol shows that most likely new strategies of
HHfrag: HMM-based fragment detection using HHpred

HHfrag-derived variable-length fragment library.

In all instances, we observed a positive shift in the distribution of
decoy TM-scores implying an increased accuracy of the predicted
structures (Fig. 8). HHfrag shifts the position of the most populated
TM-score bin and increases the fraction of good decoys (TM-score
> 0.4) by 31, 26, 14 and 29%, respectively. Although the best decoys
generated with both methods have essentially the same TM-score,
good decoys are produced 1.4–16.0 times more often when using a
fragment library built with HHfrag.

4 CONCLUSION

We have introduced HHfrag, a new HMM-based fragment detection
method that uses the profile comparison tool HHpred to build a
customized fragment library for a query protein sequence. Our
results show that a dynamic fragment library has advantages over a
static library in that it improves the sequence coverage dramatically.
Compared to other dynamic approaches such as Rosetta NNmake,
HHfrag improves the precision of the fragments significantly at the
expense of a 19.4±15% loss in sequence coverage. A distinctive
advantage is that HHfrag extracts fragments with variable length
that may also contain gaps.

Often fragments identified by HHfrag seem to point at a common
evolutionary origin of the proteins sharing the same motif. Consider
the example of the GD box (Alva et al., 2009). The GD box links
evolutionarily homologous members of the cradle-loop barrel metafold
(Alva et al., 2008) and also otherwise unrelated folds sharing an
analagous motif. HHfrag detects GD boxes with very high coverage
and precision. This example shows that fragments found by HHfrag
may arise from a common evolutionary origin and can be also the
result of convergence.

Our experiences with decoy generation using Rosetta’s
AbInitioRelax protocol shows that most likely new strategies of
fragment assembly need to be developed in order to take full advantage of the HHfrag approach. Tests with an ideal fragment library show that even few gaps in the fragment map may have a disastrous effect on the distribution of decoys (Supplementary Material). Moreover, an increased number of false positives slows down the convergence and can also increase the population of misfolded decoys. In the examples where Rosetta successfully generates near-native decoys, we see a clear enrichment of near-native structures when using HHfrag fragments. Often, this enrichment is the result of a more pronounced folding funnel (see energy versus TM-score plots in the Supplementary Material).

Here, our main focus is on how to select fragments that capture local protein structure and not on how to improve the Rosetta structure prediction protocol. Fragments are useful for many purposes, not only for structure prediction. Fragment-based approaches have, for example, been instrumental in the recent structure determination of mitochondrial uncoupling protein 2 (Berardi et al., 2011). Future work will focus on the combination of HHfrag fragments with sparse and low-quality experimental data.

Funding: Deutsche Forschungsgemeinschaft (DFG) grant HA 5918/1-1, by contract research ‘Methoden in den Lebenswissenschaften’ of the Baden-Württemberg Stiftung and by the Max Planck Society.

Conflict of Interest: none declared.

REFERENCES