Formatt: Correcting Protein Multiple Structural Alignments by Sequence Peeking

Shilpa Nadimpalli* Noah Daniels* Lenore Cowen†
Department of Computer Science, Tufts University
161 College Ave, Medford, MA 02155.

ABSTRACT

We present Formatt, a multiple structure alignment program based on the Matt purely geometric multiple structural alignment program, that also takes into account sequence similarity when constructing alignments. We show that Formatt is superior to Matt in alignment quality based on objective measures (most notably Staccato sequence and structure scores) while preserving the same advantages in core length and RMSD that Matt has as a flexible structure aligner, as compared to other multiple structure alignment programs on popular benchmark datasets. Applications include producing better training data for threading methods.

1. INTRODUCTION

A classical problem in computational biology is to construct structural alignments of multiple homologous proteins. Typically, both the protein sequence and its 3D structure are available to a structural alignment program. The structural alignment program typically produces both a rigid body transformation that aligns the structures in space, plus a sequence alignment derived from that structural alignment that proposes homologous residue-residue correspondences. For a recent survey of the best current structural alignment programs available, see [4]. In the absence of hand-curated gold-standard benchmarks, the quality of protein structure alignment is usually measured based on purely geometric measures: some function of the number of residues declared to be alignable, together with an average RMSD score for aligned residues, plus perhaps a penalty for gaps. Similarly, most of the best structural alignment programs in use today begin by ignoring all sequence information, and working only with the geometric location of the C_α atoms of the protein backbones. It seems that this extra information could be used to improve protein structural alignment. However, a meaningful way to incorporate sequence information into structural alignment algorithms in order to improve their performance has remained elusive.

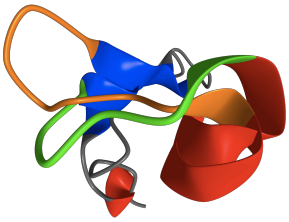
One of the reasons it has not been clear how best to incorporate sequence information into structural alignment programs is that it is unclear what the goal is, or rather, the goal might be problem-dependent. When a sequence alignment and a structure alignment of two protein sequences give different answers, which one is correct? If the correct alignment is defined solely based on the geometric location of the C_α atoms of the protein backbones, then this alignment can always be computed without ever looking at the protein sequences. At the opposite end of the spectrum, we could imagine a "true" correct alignment to be one that aligns residues that have evolved from residues in a common ancestor protein. Ignoring the fact that constructing a gold-standard benchmark to test alignment algorithms according to this standard would be impossible without time travel, such an alignment might result in aligned regions with very little geometric similarity. Should these regions still be considered alignable?

Several researchers have developed algorithms, including 3DCoffee [15], PROMALS3D [17], and SALIGN [9], that consider both sequence and structure when constructing protein alignments. As has been demonstrated by Kim and Lee [6], structure-based methods produce better sequence alignments than methods based on sequence information alone. These algorithms have all, to some extent, had to address the question of what their hybrid algorithm considers a "correct" alignment. However, with the notable exception of SALIGN (see below) most of these papers try to use *structural* information to improve *sequence* alignments, whereas the goal of this paper is to use *sequence* information to improve *structural* alignments. Even though the "correct" alignment in both scenarios is presumably the same, these are two very different problems, because the natural assumptions on the inputs to the two problems are completely different: i.e., sequence alignment programs cannot assume structural information is available for all proteins.

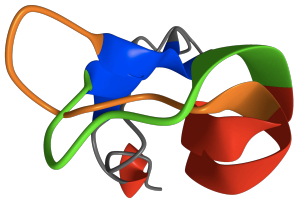
Instead of asking if (partial) structural information can help sequence alignment algorithms, this paper instead focuses on what we believe is a substantially easier computational problem: we ask if sequence information can help structural alignment algorithms in the typical setting where purely structural alignment algorithms are employed, specifically when 3D structural information is available for all the proteins in the set. We suspected it would help, because anecdotally, for even the best structural alignment programs, we knew there were always cases where it seemed

*These authors contributed equally to this work
†Corresponding author: cowen@cs.tufts.edu

ACM-BCB '11, August 1-3, Chicago, IL, USA



(a) Matt structural alignment



(b) Formatt structural alignment

CVR–FQL–PMPGSRLC
LETLLLNGVL–––TLV

(c) Matt sequence alignment

CVRFQLPMPGS–RLC
LET––LLLNGVLTLV

(d) Formatt sequence alignment

Figure 1: Example of Formatt's frame-offset repair on a subset (residues 37-50 of chain A of PDB ID 1c9f, and residues 64-76 of chain A of PDB ID 1d4b) of the HOMSTRAD "CIDE-N" group. In both sequence and structural alignments, difference between Matt and Formatt are shown in orange and green; red and blue regions are α and β structures aligned identically by Matt and Formatt. Note that the Formatt alignment has fewer non-core residues (three) than Matt (five).

a human being could hand-"correct" the alignment into something that made more sense from a sequence point of view, with little or no loss in geometric fidelity. The kinds of errors produced by structure alignment programs that do not take sequence into account can be illustrated by an example pair of proteins, aligned by our group's own structure alignment program, Matt [10]. Figure 1 illustrates how the structural alignments produced are quite similar, but the Formatt sequence alignment has fewer gaps, and thus fewer non-core residues (three) than Matt (five). Note that while we have chosen to show a bad alignment produced by our Matt program, all the other purely structural alignment algorithms that we have tested will sometimes produce similar types of errors.

To avoid these offset problems, we modify Matt to "peek" at the sequence in order to correct this type of register error. In particular, we introduce "Formatt" which stands for "Frame Offset Repair Matt" which uses the same geometric information that Matt uses to decide what regions of the protein should be considered alignable. Formatt allows Matt to construct its bent alignment, which breaks a protein up into small tightly aligned blocks, between which are regions where Matt would greedily align the backbone between blocks (the Matt "extension phase") using solely geometric criteria. Formatt, by contrast, considers both geometric and sequence similarity criteria in choosing which residues to align in these regions.

Note that our Matt structural aligner is specifically optimized for more distant homology [1] and as we find again in this paper, classical aligners may perform better on highly homologous sequences. However, the hope is the Formatt correction will improve Matt performance on closely homologous sequences while preserving Matt's performance advantage on remote homologs. We show below that this is indeed the case.

We test the performance of Formatt against the original Matt [10], against Mustang [7], another well-known multiple structure alignment program, and against SALIGN [9], which like Formatt incorporates sequence information into a structural alignment. We also considered 3DCoffee [14] and Promals3D [17] but found they were not competitive even on the closely-aligned structures in the popular HOMSTRAD benchmark. Of course, as remarked above, to be fair to 3DCoffee and Promals3D, they can also produce alignments (which Formatt cannot) when structural information is only available for a subset of the protein sequences to be aligned, and were not optimized for the full-information structural alignment problem.

The metrics under which we tested performance on HOMSTRAD include the correct gold-standard reference alignments (which were curated by hand). On the SABMark "Twilight Zone" benchmark [22], which we chose to capture the alignment of more remotely homologous proteins, there is no gold-standard reference, and so another measure of alignment quality must be devised. We show that Formatt alignments are superior to Matt alignments according to a purely objective measure that does not require a reference alignment; namely, the "Staccato" scores as introduced by Shatsky, Nussinov and Wolfson [20]. While Mustang and SALIGN both produce reasonable HOMSTRAD alignments, and in fact their HOMSTRAD alignments match the reference alignments slightly better than either Matt or Formatt, neither Mustang nor SALIGN produce SABMark alignments with reasonable RMSD, in contrast to both Formatt and Matt.

Formatt source code is freely available for download under the Gnu Public License at http://bcb.cs.tufts.edu/formatt where we also make available HOMSTRAD and SABMark benchmark reference alignments aligned by Formatt.

2. METHODS

2.1 Matt

The Matt structural aligner [10] belongs to the class of fragment-pair chaining method aligners. Matt finds blocks of between 5 and 9 amino acids in each chain participat-

ing in a multiple alignment that share close spatial alignment, without regard to the fact that the regions between these blocks may include impossible bends, translations, or twists. Matt then extends these aligned blocks, adding adjacent amino acids that do not diverge greatly in spatial alignment. Thus, Matt aligns protein sequences based on root mean square distance (RMSD). Matt solves a bi-criterion optimization problem, balancing the length of the aligned cores with the minimization of RMSD. This balance was achieved by finding a linear combination of RMSD and core length that optimally separated SABMark [22] positive from decoy chains at the superfamily level of homology.

2.2 Improving upon Matt

The chief limitation of Matt's approach is that the regions in between the original, closely-aligned, 5-9 amino acid blocks are still aligned purely according to this balance between core length and RMSD, and thus the final alignment may choose arbitrarily between different possible alignments of similar RMSD values. This can lead to otherwise obvious sequence similarities being discarded due to negligible differences in RMSD. By preserving sequence information, and allowing the input from a pure sequence alignment tool to influence the final alignment, we aim to improve the alignments of these regions between closely-aligned blocks.

Formatt produces an initial "bent" alignment of 5-9 amino acid blocks, identically to Matt. It then extends each aligned block as follows: given a region of up to 20 residues between blocks, we measure the optimal RMSD of the region for all possible alignments of this inter-block region. If this RMSD is less than Matt's original spatial alignment threshold (5 Å), a sequence aligner is run instead and the resulting multiple sequence alignment is used for the inter-block region (we currently use MUSCLE [3], but any multiple sequence alignment application could be used). If the RMSD of this inter-block region is greater than 5 Å, or the region is longer than 20 residues, the region is aligned greedily based on RMSD.

The choice of window size to pass to the sequence aligner determines when sequence rather than RMSD-based structure alignment is used for inter-block regions. Inter-block regions longer than the window size are greedily aligned based purely on RMSD. We present results for window size choices of 5, 10, 15, and 20 residues.

2.3 Validation

In order to quantitatively assess Formatt's performance, we evaluate it against two well-known benchmark sets, HOMSTRAD [12] and SABMark [22].

The HOMSTRAD multiple-alignment benchmark consists of a manually curated set of 1,028 alignments, each of which contains between two and 41 structures. We primarily test on the 398 HOMSTRAD alignments with more than two structures in the alignment (that is, HOMSTRAD sets with between three and 41 structures that necessitate a multiple rather than a pairwise structure alignment program). For HOMSTRAD alignments, we can assume the manually curated alignment form a gold-standard set of "correct" alignments.

The SABMark benchmark is divided into superfamily and "Twilight Zone" benchmark datasets, each of which contains subsets of 3 to 25 remotely homologous protein structures. We test Formatt and its competitors on the 209 subsets in the "Twilight Zone" set. Note that for these more distant homologs, we do not have a gold-standard set of "correct" alignments, and must determine alignment quality by objective means, such as core length, average pairwise RMSD, as well as the Staccato scores, as introduced by [20]. Details on how to compute each score can be found in [20]. We diverge from the Staccato paper in the way that we compute these scores in one important respect: we only consider core positions in the alignment (where a core position places no gaps in the alignment) when scoring a multiple alignment.

The Staccato "Seq" score measures sequence alignment quality and is a normalized sum-of-pairs score based upon the BLOSUM62 matrix. The Staccato "Str" score is a measure of what percentage of core residues in an alignment have an RMSD of < 3 Å, with core residues defined as those columns in the multiple alignment for which every chain has a residue, or equivalently, those columns without gaps.

3. RESULTS

Table 1: HOMSTRAD Multiple Alignments

	Avg. core	Avg. RMSD	Avg. % Correct
HOMSTRAD	126.8	2.71	(100%)
Mustang	152.8	3.60	**79.3%**
Matt	178.4	**1.72**	73.4%
SALIGN	172.6	2.29	78.1%
Formatt (5)	178.6	1.79	73.5%
Formatt (10)	178.9	1.83	73.4%
Formatt (15)	179.4	1.93	73.4%
Formatt (20)	**179.6**	1.95	73.4%

As can be seen in Table 1, on the 398 HOMSTRAD multiple alignments, all the aligners do a reasonable job on this benchmark. The HOMSTRAD gold standard has the smallest average core length, followed by Mustang and SALIGN. Matt's average core length is longer, and each version of Formatt with progressively longer sequence alignment windows achieves a progressively longer core length. On the other hand, as the Formatt alignment window increases, RMSD increases slightly compared to Matt as well, as would be expected. Formatt's and Matt's percent correct, however, according to the gold standard, underperforms the other methods, though the difference between Formatt and Matt by this measure is negligible. Thus, we conclude that Formatt and Matt by the most objective measure (both core length and RMSD) outperform other methods on the HOMSTRAD benchmark set, but underperform them according to the HOMSTRAD hand-curated alignments. This may be inevitable when Matt and Formatt increase the size of the core, as the gold standard alignment appears to be conservative. On the other hand, it is clear that in comparing Formatt to Matt, Formatt increases core length for a small penalty in RMSD.

We also tested Promals3D on the HOMSTRAD benchmark set. Note that Promals3D outputs only a sequence alignment without coordinates, so an RMSD was not calculated. However, when we compared the Promals3D to the HOMSTRAD gold-standard alignments, the average percentage correct was only 18.6%. We tested a subset of the HOMSTRAD benchmark set against 3DCoffee and the results were even worse. Thus, we conclude that Promals3D

and 3DCoffee are not producing competitive alignments on this benchmark.

Table 2: SABMark Twilight Zone

	Avg. core length	Avg. RMSD
Mustang	63.4	11.83
Matt	**66.9**	**2.64**
SALIGN	59.6	22.15
Formatt (5)	**66.9**	**2.64**
Formatt (10)	**66.9**	2.68
Formatt (15)	66.8	2.71
Formatt (20)	66.8	2.74

Table 2 shows that Matt and Formatt outperform Mustang and SALIGN both in terms of core length and RMSD on the SABMark "Twilight Zone" benchmark set. Here, however, we do not have gold-standard reference alignments. It is also less immediately clear from Table 2 whether Formatt or Matt alignments are to be preferred. To further study this question, we look at the Staccato scores suggested by [20] on both HOMSTRAD and SABMark. Figure 2 shows that on HOMSTRAD, Formatt with a window size of 5, for example, has a Staccato "Seq" score and "Str" score greater than or equal to Matt on 242 out of 398 groups, and a "Seq" score greater than or equal to Matt but a "Str" score less than Matt on 95 groups. Figure 3 shows that on SABMark's "Twilight Zone" set, Formatt with a window size of 5 has a Staccato "Seq" score and "Str" score greater than or equal to Matt on 175 out of 209 groups, and a "Seq" score greater than or equal to Matt but a "Str" score less than Matt on 21 groups. Thus, by the objective Staccato measures, Formatt's alignments are superior to Matt's alignments. As Formatt window size increases, the Staccato "Seq" score goes up in a tradeoff against the Staccato "Str" score.

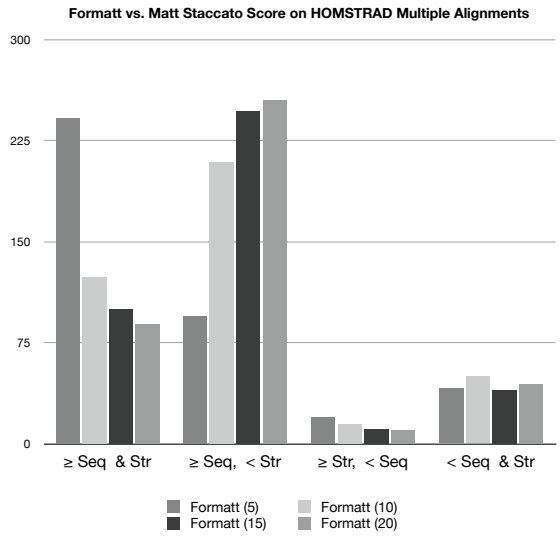


Figure 2: Histogram showing the effect of different Formatt window lengths on performance vs. Matt, on the HOMSTRAD multiple alignments. For the most HOMSTRAD groups, Formatt with a window size of 5 outperforms Matt on the Staccato "Seq" score, while rarely performing worse on the "Str" score. As window size increases, Formatt outperforms Matt more frequently on "Seq" score but less frequently on "Str" score.

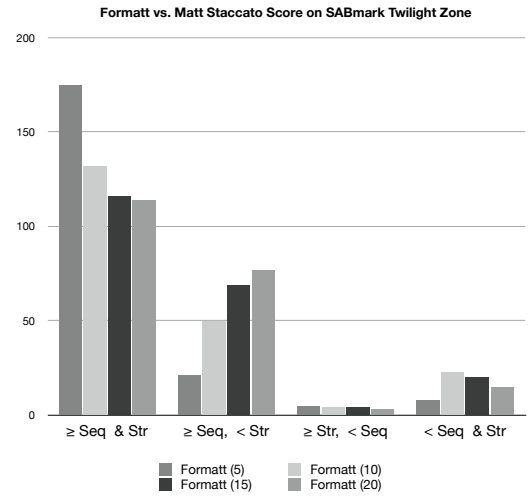


Figure 3: Histogram showing the effect of different Formatt window lengths on performance vs. Matt, on the SABMark Twilight Zone set. For the most SABMark groups, Formatt with a window size of 5 outperforms Matt on the Staccato "Seq" score, while rarely performing worse on the "Str" score. As window size increases, Formatt outperforms Matt more frequently on "Seq" score but less frequently on "Str" score.

4. DISCUSSION

We have introduced Formatt and showed that incorporating sequence information can improve the quality of structural alignments, both in terms of gold-standard alignment benchmarks, and in terms of objective measures of sequence and structural alignment quality such as the Staccato score [20]. We were particularly interested in "correcting" Matt structural alignments to better capture sequence homology because of our extensive use of the Matt structural alignment program in the training phase as we build HMMs [8] and Markov Random Fields [11] from sets of solved protein structures that fold into the similar shapes, to learn to recognize new protein sequences that match these models. More consistent alignments lead to better structural templates, and therefore better motif recognition programs. This is the same problem domain that motivated the work on the SALIGN program as well [9].

Formatt is a variant of the Matt [10] multiple structure alignment program, one of a new generation of structural alignment programs that incorporate flexibility into multiple protein structure alignments. Other recent pairwise and multiple structure alignment programs that also incorporate some form of flexibility into alignments include FlexProt [19], Fatcat [23], Posa [24], Rapido [13], and FlexSnap [18]. It would be interesting to see if some form of sequence alignment could be incorporated into these programs as well, and whether it could improve their structural alignments.

As mentioned above, our current implementation of Formatt was tested using only two different popular sequence alignment methods, namely CLUSTAL-W [21] and MUSCLE [3]. We only reported results for MUSCLE, since they were far superior to results with CLUSTAL-W. However, many newer multiple sequence alignment programs have recently been shown to perform well on more distantly homologous sequences, such as ProbCons [2], MUMMALS [16] and MAFFT [5]. It would be interesting to see if substituting some of these programs for MUSCLE in Formatt's multiple sequence alignments would improve Formatt results still further. Note that the optimal window size where Formatt would realign Matt alignments might be sensitive to choice of sequence aligner. Indeed, since the optimal window size is unlikely to be constant, we hope to introduce further refinements to Formatt by choosing, for each inter-block region, the sequence-based or RMSD-based alignment that optimizes a combined Staccato "Cons" score [20].

5. ACKNOWLEDGEMENTS

Thanks to Matt Menke for expert help with the Matt codebase. This work was funded in part by NIH grant 1R01GM080330-01A1 (to L.C.).

6. REFERENCES

[1] N. Daniels, A. Kumar, L. Cowen, and M. Menke. Touring protein space with Matt. *Bioinformatics Research and Applications*, 6053/2010:18–28, Jan 2010.

[2] C. Do, M. Mahabhashyam, M. Brudno, and S. Batzoglou. Probcons: probabilistic consistency-based multiple sequence alignment. *Genome Research*, 15:220–240, 2005.

[3] R. Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.*, 32:1792–1797, 2004.

[4] H. Hasegawa and L. Holm. Advances and pitfalls of protein structural alignment. *Current Opinion in Structural Biology*, 19(3):341 – 348, 2009.

[5] K. Katoh and H. Toh. Recent developments in the MAFFT multiple sequence alignment program. *Briefings in Bioinformatics*, 9:286–298, 2008.

[6] C. Kim and B. Lee. Accuracy of structure-based sequence alignment of automatic methods. *BMC Bioinformatics*, 8:355, 2007.

[7] A. Konagurthu, J. Whisstock, P. Stuckey, and A. Lesk. MUSTANG: A multiple structural alignment algorithm. *Proteins: Structure, Function, and Bioinformatics*, 64:559–574, 2006.

[8] A. Kumar and L. Cowen. Recognition of beta structural motifs using hidden Markov models trained with simulated evolution. *Bioinformatics*, 26:i287–i293, 2010.

[9] M. Madhusudhan, B. M. Webb, M. A. Marti-Renom, N. Eswar, and A. Sali. Alignment of multiple protein structures based on sequence and structure features. *Protein Engineering, Design and Selection*, pages 1–6, 2009.

[10] M. Menke, B. Berger, and L. Cowen. Matt: Local flexibility aids protein multiple structure alignment. *PLoS Computational Biology*, 4(1):e10, 2008.

[11] M. Menke, B. Berger, and L. Cowen. Markov random fields reveal an N-terminal double propeller motif as part of a bacterial hybrid two-component sensor system. *PNAS*, 107:4069–4074, 2010.

[12] K. Mizuguchi, C. Deane, T. L. Blundell, and J. Overington. HOMSTRAD: a database of protein structure alignments for homologous families. *Protein Sci.*, 11:2469–2471, 1998.

[13] R. Mosca, B. Brannetti, and T. R. Schneider. Alignment of protein structures in the presence of domain motions. *BMC Bioinformatics*, 9:352, 2008.

[14] C. Notredame, D. Higgins, and J. Heringa. T-coffee: a novel method for fast and accurate multiple sequence alignment. *Journal of Molecular Biology*, 302(1):205 – 217, 2000.

[15] O. O'Sullivan, K. Suhre, C. Abergel, D. Higgins, and C. Notredame. 3DCoffee: combining protein sequences and structures with multiple sequence alignments. *Journal of Molecular Biology*, 340:385–395, 2004.

[16] J. Pei and N. Grishin. MUMMALS: multiple sequence alignment improved by using hidden Markov models with local structure information. *Nucleic Acids Res.*, 34:4364–4374, 2006.

[17] J. Pei, B. H. Kim, and N. V. Grishin. PROMALS3D: a tool for multiple protein sequence and structure alignments. *Nucleic Acids Res.*, 36:2295–2300, 2008.

[18] S. Salem, M. J. Zaki, and C. Bystroff. FlexSnap: Flexible non-sequential protein structure alignment. *Algorithms in Molecular Biology*, 12(5), 2010.

[19] M. Shatsky, R. Nussinov, and H. Wolfson. Flexible protein alignment and hinge detection. *Proteins*, 48:242–256, 2002.

[20] M. Shatsky, R. Nussinov, and H. Wolfson. Optimization of multiple-sequence alignment based on multiple-structure alignment. *Proteins: Structure, Function and Bioinformatics*, 62:209–217, 2006.

[21] J. D. Thompson, D. G. Higgins, and T. J. Gibson. CLUSTAL-W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, 22:4673–4680, 1994.

[22] I. VanWalle, I. Lasters, and L. Wyns. SABmark–a benchmark for sequence alignment that covers the entire known fold space. *Bioinformatics*, 21:1267–1268, 2005.

[23] Y. Ye and A. Godzik. Flexible structure alignment by chaining aligned fragment pairs allowing twists. *Bioinformatics*, Suppl 2:II246–II255, 2003.

[24] Y. Ye and A. Godzik. Multiple flexible structure alignment using partial order graphs. *Bioinformatics*, 21:2362–2369, 2005.