A NEW APPROACH TO SUBOPTIMAL PAIRWISE SEQUENCE ALIGNMENT

Peter Clote
Laboratoire de Recherche en Informatique (LRI), Univ Paris-Sud, 91405 Orsay Cedex, France.
Laboratoire d'Informatique (LIX), Ecole Polytechnique, 91128 Palaiseau, France.
Department of Biology, Boston College, Chestnut Hill, MA 02467, USA.
Email: clote@bc.edu
Feng Lou
Laboratoire de Recherche en Informatique (LRI), Univ Paris-Sud, 91405 Orsay Cedex, France.
CNRS, France
INRIA Saclay, AMIB project, France
Email: lou@lri.fr
Alain Denise
Laboratoire de Recherche en Informatique (LRI), Univ Paris-Sud, 91405 Orsay Cedex, France.
Institut de Génétique et Microbiologie (IGM), Univ Paris-Sud, 91405 Orsay Cedex, France.
CNRS, France
INRIA Saclay, AMIB project, France
Email: alain.denise@lri.fr

ABSTRACT
In comparative protein modeling, the quality of a template model depends heavily on the quality of the initial alignment between a given protein with unknown structure to various template proteins, whose tertiary structure is available in the Protein Data Bank (PDB). Although pairwise sequence alignment has been solved for more than three decades, there remains a large discrepancy between the accuracy of the best *sequence* alignment between two amino acid sequences, as produced by the Needleman-Wunsch [15] or Smith-Waterman [19] algorithms, and that of the best *structural* alignment between two protein X-ray structures, as produced by the software DALI, CE, Topofit, etc. To improve the quality of initial alignments in template modeling, one can integrate valuable information from an ensemble of generated suboptimal alignments, that is alignments whose score is below the best possible score. In this paper, we present a novel algorithm to produce suboptimal pairwise alignments.

Specifically, given any initial alignment $\mathbb{A}_0$ of two nucleic acid or amino acid sequences, our algorithm SubOpt *simultaneously* computes the optimal alignment $\mathbf{A}_k$ having trace distance k from $\mathbb{A}_0$, thus producing a small, representative yet divergent ensemble of suboptimal structures in time and space $O(n^3)$. A web server for SubOpt is under construction at http://bioinformatics.bc.edu/clotelab/.

KEY WORDS
Key words: pairwise alignment, suboptimal alignment, BAliBASE.

1 Introduction

There is a well-known, demonstrable gap between the quality of sequence alignments and that of structure alignments, when benchmarked with hand-curated structure alignments of protein X-ray structures from the Protein Data Bank [5]. In particular, using structures in the SCOP (Structural Classification of Proteins) database [4] as the golden standard, Sauder et al. [16] showed that at the level of 10-15% sequence identity, BLAST [3] correctly aligns 28% of residue pairs, while PSI-BLAST [2, 1] improves the per-residue alignment accuracy to 40%. Sauder et al. [16] also showed that structure alignments of the protein tertiary structures, as produced by CE [17] and DALI [9], correctly align 75% of residue pairs at the same level of 10-15% sequence identity.

A number of groups have introduced new ideas in the attempt to identify pairs a_i, b_j of aligned residues from two amino acid sequences $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$ in an optimal sequence alignment. The overall idea is that if a_i, b_j occur in many *suboptimal* alignments, then it is likely that a_i, b_j are "correctly" aligned – i.e. a_i, b_j are more likely to be aligned in the structural alignment, as produced by DALI [9], CE [17], TOPOFIT [10], etc. (For benchmarking purposes, following the work of [16, 18], we identify the "correct" alignment with that produced by a structural alignment algorithm, such as DALI.) Thus the main question is no longer "What is the best alignment?", but rather "What are the good alignments?" (paraphrased from [21]).

Waterman [23] was the first to consider the problem of generating suboptimal alignments. In [23], he described how to modify the standard recursive traceback algorithm, in order to generate all alignments whose score exceeds a user-defined threshold. The problem with this method in

practice is that there is an enormous number of suboptimal structures, which deviate very slightly from the optimal alignment and whose score is almost as large as that of the optimal alignment. In [24], Waterman and Eggert described how to generate a first suboptimal (local) alignment produced by not allowing the alignment of any residue pair found in the optimal alignment; a second suboptimal (local) alignment is produced by not allowing any residue pair from the optimal and first suboptimal alignment, etc. In [25], Waterman et al. described how to efficiently find all optimal alignments for all choices of the penalty parameters (gap initiation, gap extension).

In [22], Vingron and Argos studied the regions of agreement between all suboptimal alignments whose score is, at most, within ϵ of that of the optimal alignment. In particular, they showed that residue pairs a_i, b_j found in all suboptimal alignments are more likely to be correctly aligned in the protein tertiary structures. In [11], Mevissen and Vingron demonstrated the utility of an edge reliability index called *robustness*, defined as the difference between the sequence alignment score for an alignment including a given pair a_i, b_j of aligned residues and the highest score for an alignment that does not include that pair.

Given amino acid sequences $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$, it is possible to compute the Boltzmann probability $p_{i,j}$ that a_i is aligned to b_j, as discovered by several groups independently [12, 13, 6]. In particular, Miyazawa [12] identified pairs a_i, b_j of residues, whose Boltzmann probability of being aligned is high, while Mückstein et al [13] sampled near-optimal alignments from the Boltzmann ensemble. Following [6, 7], the main idea can be summarized as follows.

Let FZ be the *forward partition* function, defined for $0 \leq i \leq n$ and $0 \leq j \leq m$ by

$$FZ(i,j) = \sum_A e^{score(A)/C}$$

where A ranges over all possible alignments of $a_1, \ldots, a_i$ with $b_1, \ldots, b_j$, $score(A)$ is the alignment score using standard BLAST parameters, and C is a constant. For simplicity, we consider a linear gap penalty of $x \cdot g$ for a gap of size x; however, it is straightforward [12, 13, 6, 7] to extend the algorithm for an affine gap penalty, by using auxiliary matrices (Gotoh's trick [8]). The similarity (BLOSUM or PAM) between amino acids a_i, b_j is denoted $sim(a_i, b_j)$.

Algorithm 1 (Forward partition function)
For $1 \leq i \leq n$ and $1 \leq j \leq m$, define $FZ(i,0) = e^{\frac{i \cdot g}{C}}$, $FZ(0,j) = e^{\frac{j \cdot g}{C}}$, and define $FZ(i,j)$ by

$$FZ(i-1,j-1) \cdot e^{\frac{sim(a_i,b_j)}{C}} + FZ(i,j-1) \cdot e^{\frac{g}{C}} + FZ(i-1,j) \cdot e^{\frac{g}{C}}.$$

Analogously, we compute the *backward partition function* BZ, defined for $1 \leq i \leq n+1$ and $1 \leq j \leq m+1$ by

$$BZ(i,j) = \sum_A e^{score(A)/C}$$

where A ranges over all possible alignments of $a_i, \ldots, a_n$ with $b_j, \ldots, b_m$.

Algorithm 2 (Backward partition function)
For $n+1 \geq i \geq 1$ and $m+1 \geq j \geq 1$, let $BZ(i,m+1) = e^{\frac{(n+1-i) \cdot g}{C}}$, $BZ(n+1,j) = e^{\frac{(m+1-i) \cdot g}{C}}$ and define $BZ(i,j)$ to be

$$BZ(i+1,j+1) \cdot e^{\frac{sim(a_i,b_j)}{C}} + BZ(i,j+1) \cdot e^{\frac{g}{C}} + BZ(i+1,j) \cdot e^{\frac{g}{C}}.$$

One can easily check that $FZ(n,m) = BZ(1,1)$ and that this value is $\sum_A e^{\frac{score(A)}{C}}$, where A ranges over all alignments of $a_1, \ldots, a_n$ with $b_1, \ldots, b_m$. The Boltzmann probability $Pr[(a_i, b_j)]$ that a_i will be aligned with b_j is then

$$\frac{FZ(i-1,j-1) \cdot e^{\frac{sim(a_i,b_j)}{C}} \cdot BZ(i+1,j+1)}{FZ(n,m)}.$$

In [18], Sierk et al. investigate which combination of features, either singly or in combination, best allows one to identify residue pairs a_i, b_j which occur in structural alignments produced by DALI and CE. In particular, these authors benchmarked *(i)* robustness, defined earlier, with *(ii)* Boltzmann pair probabilities, with *(iii)* the maximum number of bits per position. Sierk et al. determined that the best method is a *logistic regression* model using all three parameters.

In this paper, we describe a novel method to produce suboptimal alignments. Moreover, we provide some preliminary results by testing our algorithm against the *BAliBASE* database [20] of manually curated (structure) alignments, created to allow the benchmarking of various sequence alignment methods.

2 Method

Let $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{b} = (b_1, \ldots, b_m)$ be sequences of nucleic acids or amino acids, and let $\mathbb{A}_0$ be an initial alignment. The alignment $\mathbb{A}_0$ is arbitrary; i.e. $\mathbb{A}_0$ can be produced by BLAST [3], the Needleman-Wunsch algorithm [15], the Smith-Waterman algorithm [19], any manually produced partial alignment, or even the *empty* alignment (i.e. having empty trace).

Recall that the *trace* $tr(\mathbb{A})$ of alignment $\mathbb{A}$ is the set of positions (i,j), where $1 \leq i \leq n$, $1 \leq j \leq m$ and a_i is aligned with b_j. Define the *trace distance* $d(\mathbb{A}, \mathbb{B})$ between alignments $\mathbb{A}$ and $\mathbb{B}$ of sequences $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{b} = (b_1, \ldots, b_m)$ to be the number of aligned residue pairs a_i, b_j where $\mathbb{A}$ and $\mathbb{B}$ differ; i.e. $|\,(\mathbb{A} \setminus \mathbb{B}) \cup (\mathbb{B} \setminus \mathbb{A})\,|$.

Dynamic programming

In this section, we consider global[1] pairwise alignment with a linear gap penalty[2] g, where the similarity score of

[1] Our program, SubOpt, can also compute near-optimal local alignments; however, this will not be discussed in the current paper.

[2] For simplicity, the pseudocode of our algorithm is given for the linear gap penalty $G(k) = k \cdot g$, for size k gap. However, our implementation

sequence elements a_i and b_j is $\sigma(a_i, b_j)$. Given sequences $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$, and given a non-negative integer $0 \leq K \leq n + m$, we define an $(n + 1) \times (m + 1) \times (K + 1)$ matrix M, where $M(i, j, k)$ is the *maximum* similarity over all pairwise alignments between $a_1, \ldots, a_i$ and $b_1, \ldots, b_j$, whose trace distance from the restriction of initial alignment $\mathbb{A}_0|_{[i,j]}$ to $a_1, \ldots, a_i$ and $b_1, \ldots, b_j$ is exactly k. After computing all values of $M(i, j, k)$ for $0 \leq i \leq n$, $0 \leq j \leq m$, and $0 \leq k \leq K$, we can produce the suboptimal alignments $\mathbb{A}_k$ for each k, which differ by exactly k residue pairs a_i, b_j from the initial alignment $\mathbb{A}_0$. In the sequel, we will refer to $\mathbb{A}_k$ as the k-alignment. Note that if $\mathbb{A}_0$ is the empty alignment, then the k-alignment $\mathbb{A}_k$ consists of exactly k aligned residue pairs a_i, b_j, whose contribution is in a very precise mathematical sense the *most important*.

Finally, we remark that in a manner similar to that of [6, 7] and described above, it is straightforward to compute the *partition function*

$$Z_k = \sum_A e^{score(A)/C}$$

where A ranges over all possible alignments of $a_1, \ldots, a_n$ with $b_1, \ldots, b_m$ for which the trace distance between A and an initially given alignment $\mathbb{A}_0$ equals k, and $score(A)$ is the alignment score using standard BLAST parameters. In this fashion, we could compute a *Boltzmann density plot* Z_k/Z.

We begin by initializing M in the base case, where either $i = 0$ or $j = 0$:

$$M(i,j,k) = \begin{cases} 0 & \text{if } i = j = k = 0 \\ -\infty & \text{if } i = j = 0 \text{ and } k > 0 \\ g \cdot j & \text{if } i = 0 \text{ and } k = 0 \\ -\infty & \text{if } i = 0 \text{ and } k > 0 \\ g \cdot i & \text{if } j = 0 \text{ and } k = 0 \\ -\infty & \text{if } j = 0 \text{ and } k > 0. \end{cases}$$

We continue with the inductive case, where $1 \leq i \leq n$, $1 \leq j \leq m$, and $0 \leq k \leq K$, as described in Figure 1.

Following the insight of Gotoh [8], in our implementation, we actually define auxilliary matrices P, Q in order to compute k-suboptimal alignments for the affine gap penalty $gap(k) = \alpha + \beta(k - 1)$, consisting of a gap initiation cost α and a gap extension cost β. This results in a program, which we call SubOpt.

3 Results

In this section, we benchmark our algorithm, SubOpt, by using alignments from the *BAliBASE* database [20], a database of manually refined multiple sequence alignments specifically designed for the evaluation and comparison of sequence alignment programs. We have taken 10 multiple alignments having sequence identity less than 25%, from which 92 pairwise alignments are obtained.

adapts the insight of Gotoh [8], in order to support the *affine* gap penalty $G(k) = \alpha + (k-1) \cdot (g-1)$, thereby accounting for distinction between gap initiation and gap extension within the same run time complexity.

3.1 Comparison of suboptimal and global alignment

Table 1 gives the number of alignments found by SubOpt among 92 reference alignments, where we used the BLOSUM45 similarity matrix, gap initiation cost $\alpha = -14$ and gap extension cost $\beta = -2$. Here, for each value of k, we considered only one k-alignment; i.e. in the traceback phase, we only generated one alignment for each value of k. Table 1 shows that 15 alignments can be found by the Needleman-Wunsch algorithm [15] using trace distance $k = 0$, but SubOpt can find 18 more alignments, where trace distance from the initial alignment varies from 2 to 40. If no reference alignment is found for a given trace distance k, then the value k does not occur in the left column of Table 1.

k	number of alignments
0	15
2	3
4	3
6	2
8	2
9	2
12	1
19	1
23	1
30	1
36	1
40	1
TOTAL	15+18=33

Table 1. The number of *BAliBASE* alignments found by our program SubOpt.

Table 2 shows the number of *BAliBASE* alignments found by SubOpt, as a function of gap initiation and gap extension parameters. As above, for each value of k we considered only one k-alignment. For values $\alpha = -14$ and $\beta = -6$, we find a maximum of 35 alignments among the 92 reference alignments.

$\beta \setminus \alpha$	-2	-4	-6	-8	-10	-12	-14	-16
-1	3	9	23	31	29	30	33	31
-2	4	11	27	31	31	32	33	30
-3	3	14	28	31	33	31	31	31
-4	3	13	24	30	32	33	31	33
-5	3	11	24	27	31	32	32	33
-6	3	11	22	27	30	33	**35**	33

Table 2. The number of alignments found by our program SubOpt, as a function of gap initiation and gap extension parameters.

Table 3 shows the percentage of *BAliBASE* reference alignments, in which SubOpt outperforms the

$$M(i,j,k) = \max \begin{cases} M(i-1,j-1,k) + \sigma(a_i, b_j) & \text{if } (i,j) \in \mathbb{A}_0 \\ M(i-1,j-1,k-2) + \sigma(a_i, b_j) & \text{if } (i,-),(r,j) \in \mathbb{A}_0, \text{ for some } 1 \le r < i \\ M(i-1,j-1,k-2) + \sigma(a_i, b_j) & \text{if } (i,r),(-,j) \in \mathbb{A}_0, \text{ for some } 1 \le r < j \\ M(i-1,j-1,k-1) + \sigma(a_i, b_j) & \text{if } (i,-),(-,j) \in \mathbb{A}_0 \\ M(i,j-1,k) + g & \text{if } (-,j) \in \mathbb{A}_0 \\ M(i,j-1,k-1) + g & \text{if } (r,j) \in \mathbb{A}_0, \text{ for some } 1 \le r \le i \\ M(i-1,j,k) + g & \text{if } (i,-) \in \mathbb{A}_0 \\ M(i-1,j,k-1) + g & \text{if } (i,r) \in \mathbb{A}_0, \text{ for some } 1 \le r \le j \end{cases}$$

Figure 1. Inductive step in the dynamic programming (forward) algorithm to compute the optimal alignment score $M(i,j,k)$ over all alignments between $a_1,\ldots,a_i$ and $b_1,\ldots,b_j$ whose trace differs in k positions from the restriction of the initial alignment $\mathbb{A}_0$ to $a_1,\ldots,a_i$ and $b_1,\ldots,b_j$. Note that in case 1 and 2 above, the traceback would align a_i and b_j, while in case 3 and 4, the traceback would align b_j with a gap symbol, and in case 5 and 6, the traceback would align a_i with a gap symbol. Both time and memory requirements are clearly $O(nmK)$.

Needleman-Wunsch algorithm, as a function of trace distance k. We do not show $k = 0$ since in this case the results are the same for both algorithms. Again, for each value of k we considered only one k-alignment. For trace distance greater than 1, most of the k-alignments produced by `SubOpt` are more similar to the reference alignment, than that produced by Needleman-Wunsch. For example, when $k = 4$, 65.1% of the 4-alignments produced by `SubOpt` more closely resemble the reference alignment than does the Needleman-Wunsch optimal alignment.

k	`SubOpt`
1	50.5%
2	58.9%
3	52.1%
4	65.1%
5	53.2%
6	67.3%
7	55.8%
8	64.7%
9	55.2%
10	65.0%
11	53.7%
12	64.6%

Table 3. The percentage of reference alignments, in which `SubOpt` outperforms the Needleman-Wunsch algorithm, as a function of trace distance k.

3.2 Comparison of 3 near-optimal alignment methods

In this section, we compare near-optimal alignments generated by three different methods: *(i)* sampling from the Boltzmann ensemble by `probA` [13], *(ii)* generating all *Zuker suboptimal alignments* using the `noptalign` web server[3] which implements Zuker's method [26] of generating alignments which are optimal and contain residue pairs a_i, b_j for various possible choices of a_i, b_j, and *(iii)* our program `SubOpt`, which generates, for each integer k, *all* alignments which have maximum possible score, contingent on the requirement that their trace distance from the Needleman-Wunsch optimal alignment $\mathbb{A}_0$ equals k. (For fixed value of k, there may be many alignments, whose trace distance with $\mathbb{A}_0$ equals k, all of which have the maximum possible alignment score. Our program `SubOpt` can generate, for each value of k, *all* such possible k-alignments by straightforward modification of the traceback phase.)

3.2.1 Entropy

We took two proteins from the multiple alignment 1aboA_ref1 - reference 1 from the *BAliBASE* database [20]: 1ihvA (Swiss Prot accession P00383) and 1pht (Swiss Prot accession P27986). These proteins, from *Escherichia coli* and from *Homo sapiens* respectively, both contain an SH3 domain, consisting of five conserved β-strands [14].

Using each of the previously described methods `probA`, `noptalign`, `SubOpt`, we generated the top scoring 111 near-optimal alignments and computed for each sequence 1ihvA and 1pht, the *position-specific entropy* $H(i)$. This was done as follows.

Consider two amino acid sequences, $\mathbf{a} = a_1,\ldots,a_m$ and $\mathbf{b} = b_1,\ldots,b_m$. For a collection of near-optimal alignments of two proteins, $\mathbf{a} = a_1,\ldots,a_m$ and $\mathbf{b} = b_1,\ldots,b_m$, we compute the frequencies $f(i,k)$, for $1 \le i \le n$ and $0 \le k \le 2m+2$, defined as follows. For $j = 1,\ldots,n$, we define $f(i,2j-1)$ to be the frequency that a_i is aligned with b_j, while for $j = 2,\ldots,m$, while $f(i,2j)$ is the frequency that a_i is aligned with a gap occurring between b_{j-1} and b_j. Finally, $f(i,0)$ is the frequency that

[3] The Pearson Lab has created a web server `http://fasta.bioch.virginia.edu/noptalign` for `noptalign`.

a_i is aligned with a gap before b_0, and $f(i, 2m)$ is the frequency that a_i is aligned with a gap after b_m. For each fixed value of i_0 in $\{1, \ldots, n\}$, $f(i_0, j)$ is a probability distribution; i.e. $\sum_{j=0}^{2m} f(i_0, j) = 1$. Hence the *position-specific entropy* $H(i)$ is defined by

$$H(i) = -\sum_{j=0}^{2m} f(i_0, j) \cdot \ln f(i_0, j).$$

The entropy $H(i)$ will be low in regions where a_i is very often aligned to the same amino acid b_j, or to a gap occurring between amino acids b_{j-1} and b_j; i.e. at position i, the near-optimal alignments tend to agree on the alignment partner of a_i.

Figure 3 displays the position-specific entropy where $\mathbf{a} = a_1, \ldots, a_n$ designates the protein 1ihvA (Swiss Prot accession P00383), while $\mathbf{b} = b_1, \ldots, b_m$ designates the protein 1pht (Swiss Prot accession P27986) and the position-specific entropy where the role of the proteins is reversed; i.e. $\mathbf{a} = a_1, \ldots, a_n$ designates the protein 1pht (Swiss Prot accession P27986), while $\mathbf{b} = b_1, \ldots, b_m$ designates the protein 1ihvA (Swiss Prot accession P00383).

We computed the position-specific entropy as follows:

- Calculate the frequency that the amino acid at position i of sequence 1ihvA aligns with a particular amino acid, or with a gap between two particular amino acids of sequence 1pht. This yields $f_1(i, j)$, where i ranges from 1 to the length of sequence 1ihvA, while j ranges from 0 to twice the length of sequence 1pht.

- Calculate the frequency that the amino acid at position i of sequence 1pht aligns with a particular amino acid, or with a gap between two particular amino acids of sequence 1ihvA. This yields $f_2(i, j)$, where i ranges from 1 to the length of sequence 1pht, while j ranges from 0 to twice the length of sequence 1ihvA.

- For each position i in sequence 1ihvA, we calculate the position-specific entropy:

$$H_1(i) = -\sum_{j} f_1(i, j) \cdot \ln(f_1(i, j))$$

  and for each position j in sequence 1pht, we calculate the position-specific entropy:

$$H_2(i) = -\sum_{j} f_2(i, j) \cdot \ln(f_2(i, j))$$

Figure 3 presents the position-specific entropy of the proteins 1ihvA, resp. 1pht, according to each of three methods `probA`, `noptalign`, `SubOpt` of generating near-optimal alignments. The position-specific entropy from our program, `SubOpt`, appears to be smaller than either of the other two methods, which suggests two aspects: *(i)* There appears to be a greater diversity in the near-optimal alignments generated by `probA` and `noptalign`, than by `SubOpt`. *(ii)* In looking at the *core blocks* of the *BAliBASE* alignment containing sequences 1ihvA and 1pht, as shown in Figure 2, it appears that the location of residues in the conserved block corresponds roughly with locations having small position-specific entropy, especially with respect to our program, `SubOpt`. We now numerically quantify both of these notions.

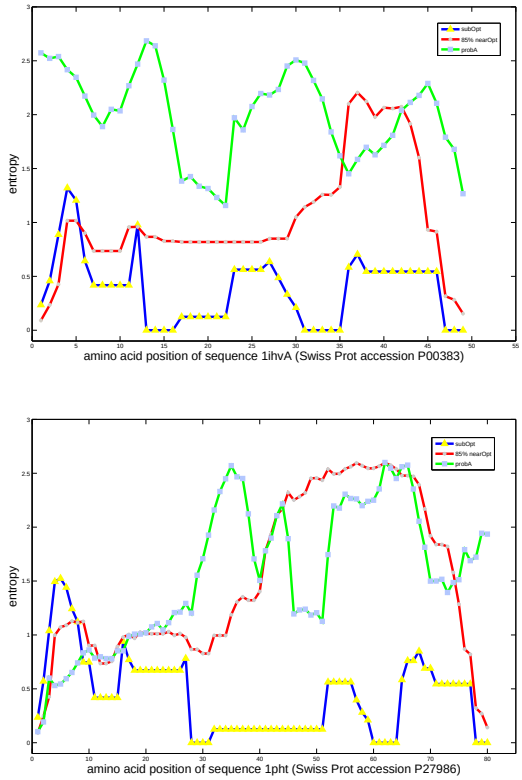

Figure 3. Position-specific entropy for near-optimal alignments generated by the three different methods. Above: first sequence (lihvA); below: second sequence (1pht).

3.2.2 Near-optimal diversity and correlation with core blocks

To numerically quantify the two previously mentioned insights, we compare the *near-optimal diversity*, defined by

$$\sum_{i=0}^{2n} \sum_{j=0}^{2m} p_{(i,j)} \cdot (1 - p_{(i,j)})$$

where $p_{2i-1,k} = f_1(i, k)$ and $p_{2i,k} = f_1(i, k)$, or equivalently, where $p_{i,2k-1} = f_2(i, k)$ and $p_{i,2k} = f_2(i, k)$. In other words, in the first sequence $\mathbf{a} = a_1, \ldots, a_n$, consider the $2n + 1$ possible *locations*, corresponding to either one of the n residues, or to a gap before a_1, or between some a_i and a_{i+1}, or after a_n. Similarly, in the second sequence

```
-NFRVYYRDsrd------pvwkGPAKLLWkg----------------eGAVVIQd--nsdiKVVPRRKAKIIRd-----
gYQYRALYDykkereedidlhlGDILTVNkgslvalgfsdgqearpeeiGWLNGYnettgerGDFPGTYVEYIGrkkisp
```

Figure 2. Core blocks from the initial segment of the *BAliBASE* alignment of two SH3 proteins 1ihvA (Swiss Prot accession P00383) and 1pht (Swiss Prot accession P27986). Aligned uppercase letters designation locations within core blocks.

$\mathbf{b} = b_1, \ldots, b_m$, consider the $2m + 1$ possible *locations*, corresponding to either one of the m residues, or to a gap before b_1, or between some b_i and b_{i+1}, or after b_m. Then $p_{i,j}$ is the frequency that the ith location from the first sequence is aligned with the jth location from the second sequence.

Additionally, we computed the Pearson correlation coefficient between positions in *core blocks* of the *BAliBASE* alignment between 1ihvA and 1pht. For each position in 1ihvA, we computed the correlation between $H(i)$ and $c(i)$, where $c(i) = 1$ if i appears in a core block of 1ihvA, and 0 otherwise (i.e. $c(i)$ is the indicator function of whether i appears in a core block of 1ihvA). This correlation is denoted *correlation 1*. Similarly, we compute *correlation 2* as the Pearson correlation between the position-specific entropy $H(i)$ of the sequence 1pht and $c(i)$, where $c(i)$ equals the indicator function of whether i appears in a core block of 1pht. Table 4 now provides a numerical quantification of the two previously mentioned insights. *(i)* There appears to be a greater diversity in the near-optimal alignments generated by `probA` and `noptalign`, than by `SubOpt`: indeed, the diversity for `probA` is 85.72, for `noptalign` is 74.4, and for `SubOpt` is 30.65. *(ii)* In looking at the *core blocks* of the *BAliBASE* alignment containing sequences 1ihvA and 1pht, one might ask whether there is any correspondence between the position-specific entropy of a location i and whether a_i is a residue in the *core block* of the *BAliBASE* alignment. Table 4 shows that correlation 1 is 0.22 for `SubOpt`, far greater than values of 0.07 for `noptalign` and -0.11 for `probA`. Similarly, correlation 2 is 0.49 for `SubOpt`, far greater than values of 0.17 for `noptalign` and -0.14 for `probA`. Based on our initial investigations, it appears that our method, `SubOpt`, is possibly more effective in indentifying likely *biologically significant* regions of an alignment. Oddly enough, this suggests that there is a significant anti-correlation between entropy with respect to `SubOpt` and whether a location belongs to a core block. We plan to investigate this phenomenon more fully with respect to many alignments in the future.

	`SubOpt`	nearOpt	probA
diversity	30.65	74.4	85.72
correlation 1	0.22	0.07	-0.11
correlation 2	0.49	0.17	-0.14

Table 4. Near-optimal alignment diversity and Pearson correlation between low entropy regions and *BAliBASE core blocks* for the three different methods.

4 Discussion

Our work presents a new algorithm concerning the analysis of protein sequences, motivated by the interest of improving the quality of pairwise sequence alignment. Mathematically optimal sequence alignments, produced by applying dynamic programming with similarity matrices (BLOSUM, PAM, etc.) do not always properly align active site residues or well-recognized structural elements. Indeed, it has long been noted that sequence alignment is substantially less accurate than structural alignment, when benchmarked against manually curated tertiary structure databases [16]. However, with the exponentially growing protein sequence databases, it remains an important bioinformatics research area to improve the quality of pairwise and multiple sequence alignments. Sierk et al. [18] have taken a step by developing a logistic regression model that exploits the notions of *robustness*, frequency of *aligned pairs* a_i, b_j, and maximum number of bits per position.

We have presented a new method of generating near-optimal pairwise global alignments. Given any initial alignment $\mathbb{A}_0$ of two nucleic acid or amino acid sequences, in cubic time our algorithm `SubOpt` simultaneously computes for all values of k the k-optimal alignment(s); i.e. those optimal alignment(s) $\mathbb{A}_k$ having trace distance k from $\mathbb{A}_0$, where $\mathbb{A}_0$ is the Needleman-Wunsch optimal alignment, or any initial alignment with which the user starts. Using the benchmark database *BAliBASE*, we have compared our algorithm `SubOpt` with the Needleman-Wunsch algorithm, showing (unsurprisingly) that the *BAliBASE* reference alignment may be closer to a (suboptimal) k-alignment than to the Needleman-Wunsch optimal alignment. More importantly, we have computed both the *diversity* and *position-specific entropy* of near-optimal alignments produced by `SubOpt`, compared with near-optimal alignments produced by `probA` [13] and `noptalign` [26]. Our method generates less diverse near-optimal alignments, yet whose position-specific entropy is more tightly (anti-) correlated with locations in the *core block* of *BAliBASE* reference alignments. For this reason, it is possible that our near-optimal residue pair alignment frequencies may lead to future improvements in sequence alignment, for instance, by integrating features from our method into that of Sierk et al. [18]. Another possible future direction is to compute near-optimal multiple sequence alignments, by extending the pairwise method presented in this paper to multiple alignments.

Acknowledgements

Research funded by the Digiteo Foundation for the project *RNAomics* and by National Science Foundation grants DMS-0817971 and DBI-0543506. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] S. F. Altschul and E. V. Koonin. Iterated profile searches with PSI-BLAST–a tool for discovery in protein databases. *Trends Biochem. Sci.*, 23(11):444–447, November 1998.

[2] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic. Acids. Res.*, 25(17):3389–3402, September 1997.

[3] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, and D.J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, October 1990.

[4] A. Andreeva, D. Howorth, S. E. Brenner, T. J. Hubbard, C. Chothia, and A. G. Murzin. SCOP database in 2004: refinements integrate structure and sequence family data. *Nucleic. Acids. Res.*, 32(Database):D226–D229, January 2004.

[5] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The protein data bank. *Nucleic Acids Researches*, 28(1):235–242, 2000.

[6] P. Clote. Biologically significant sequence alignments using boltzmann probabilities. `http://bioinformatics.bc.edu/clotelab/BoltzmannAlignment`, 2003.

[7] P. Clote and J. Straubhaar. Symmetric time warping, Boltzmann pair probabilities and functional genomics. *J. Math. Biol.*, 53(1):135–161, July 2006.

[8] O. Gotoh. An improved algorithm for matching biological sequences. *J. Mol. Biol.*, 162(3):705–708, December 1982.

[9] L. Holm and C. Sander. Dali: a network tool for protein structure comparison. *Trends Biochem. Sci.*, 20(11):478–480, November 1995.

[10] V. A. Ilyin, A. Abyzov, and C. M. Leslin. Structural alignment of proteins by a novel TOPOFIT method, as a superimposition of common volumes at a topomax point. *Protein. Sci.*, 13(7):1865–1874, July 2004.

[11] H.T. Mevissen and M. Vingron. Quantifying the local reliability of a sequence alignment. *Protein Eng.*, 9(2):127–132, 1996.

[12] S. Miyazawa. A reliable sequence alignment method based on probabilities of residue correspondences. *Protein. Eng.*, 8(10):999–1009, October 1995.

[13] U. Muckstein, I. L. Hofacker, and P. F. Stadler. Stochastic pairwise alignments. *Bioinformatics*, 18:S153–S160, 2002.

[14] A. Musacchio, T. Gibson, V. P. Lehto, and M. Saraste. SH3 – an abundant protein domain in search of a function. *FEBS Letters*, 307(1):55 – 61, 1992.

[15] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, 48(3):443–453, March 1970.

[16] J. M. Sauder, J. W. Arthur, and R. L. Dunbrack, Jr. Large-scale comparison of protein sequence alignment algorithms with structure alignments. *Proteins.*, 40(1):6–22, July 2000.

[17] I. N. Shindyalov and P. E. Bourne. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein. Eng.*, 11(9):739–747, September 1998.

[18] M. L. Sierk, M. E. Smoot, E. J. Bass, and W. R. Pearson. Improving pairwise sequence alignment accuracy using near-optimal protein sequence alignments. *BMC. Bioinformatics*, 11:146, 2010.

[19] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147(1):195–197, March 1981.

[20] J. D. Thompson, F. Plewniak, and O. Poch. A comprehensive comparison of multiple sequence alignment programs. *Nucleic. Acids. Res.*, 27(13):2682–2690, July 1999.

[21] M. Vingron. Near-optimal sequence alignment. *Curr. Opin. Struct. Biol.*, 6(3):346–352, June 1996.

[22] M. Vingron and P. Argos. Determination of reliable regions in protein sequence alignments. *Protein. Eng.*, 3(7):565–569, July 1990.

[23] M. S. Waterman. Sequence alignments in the neighborhood of the optimum with general application to dynamic programming. *Proc. Natl. Acad. Sci. U.S.A.*, 80(10):3123–3124, May 1983.

[24] M. S. Waterman and M. Eggert. A new algorithm for best subsequence alignments with application to tRNA-rRNA comparisons. *J. Mol. Biol.*, 197(4):723–728, October 1987.

[25] M. S. Waterman, M. Eggert, and E. Lander. Parametric sequence comparisons. *Proc. Natl. Acad. Sci. U.S.A.*, 89(13):6090–6093, July 1992.

[26] M. Zuker. Suboptimal sequence alignment in molecular biology. Alignment with error analysis. *J. Mol. Biol.*, 221(2):403–420, September 1991.