

Homework Assignment 1

BENG 202/CSE 282

January 30, 2007

This homework is due February, 6 2007 at 3:30 PM. This is an individual assignment. You can discuss the homework with other people in the class, but make sure you document this. Please hand in a hardcopy on the due date.

All problems were taken from N.C. Jones and P.A. Pevzner. Introduction to Bioinformatics Algorithms. The MIT Press. 2004.

Problem 6.20 (10 Points)

Consider the sequences $\mathbf{v} = \text{TACGGGTAT}$ and $\mathbf{w} = \text{GGACGTACG}$. Assume that the match premium is $+1$ and the mismatch and indel penalties are -1 .

- Fill out the dynamic programming table for a global alignment between \mathbf{v} and \mathbf{w} . Draw arrows in the cells to store the backtrack information. What is the score of the optimal global alignment and what alignment does this score correspond to?
- Fill out the dynamic programming table for a local alignment between \mathbf{v} and \mathbf{w} . Draw arrows in the cells to store the backtrack information. What is the score of the optimal local alignment in this case and what alignment achieves this score?
- Suppose we use an affine gap penalty where it costs -20 to open a gap, and -1 to extend it. Scores of matches and mismatches are unchanged. What is the optimal global alignment in this case and what score does it achieve?

Problem 6.22 (18 Points)

Define an *overlap alignment* between two sequences $\mathbf{v} = v_1 \dots v_n$ and $\mathbf{w} = w_1 \dots w_m$ to be an alignment between suffix of \mathbf{v} and a prefix of \mathbf{w} . For example, if $\mathbf{v} = \text{TATATA}$ and $\mathbf{w} = \text{AAATTT}$, then a (not necessarily optimal) overlap alignment between \mathbf{v} and \mathbf{w} is

```
ATA
AAA
```

Optimal overlap alignment is an alignment that maximizes the global alignment score between v_i, \dots, v_n and w_1, \dots, w_j , where the maximum is taken over all suffixes v_i, \dots, v_n of \mathbf{v} and all prefixes w_1, \dots, w_j of \mathbf{w} .

Give an algorithm which computes the optimal overlap alignment, and runs in time $O(nm)$.

Problem 6.23 (18 Points)

Suppose that we have sequences $\mathbf{v} = v_1 \dots v_n$ and $\mathbf{w} = w_1 \dots w_m$, where \mathbf{v} is longer than \mathbf{w} . We wish to find a substring of \mathbf{v} which best matches *all* of \mathbf{w} . Global alignment won't work because it would try to align all of \mathbf{v} . Local alignment won't work because it may not align all of \mathbf{w} . Therefore this is a distinct problem which we call the *Fitting Problem*. *Fitting* a sequence \mathbf{w} into a sequence \mathbf{v} is a problem of finding a substring \mathbf{v}' of \mathbf{v} that maximizes the score of alignment $s(\mathbf{v}', \mathbf{w})$ among all substrings of \mathbf{v} . For example, if $\mathbf{v} = \text{GTAGGCTTAAGGTTA}$ and $\mathbf{w} = \text{TAGATA}$, the best alignment might be

	global	local	fitting
\mathbf{v}	GTAGGCTTAAGGTTA	TAG	TAGGCTTA
\mathbf{w}	-TAG----A---T-A	TAG	TAGA--TA
score	-3	3	2

The scores are computed as 1 for match, -1 for mismatch or indel. Note that the optimal local alignment is not a valid fitting alignment. On the other hand, the optimal global alignment contains a valid fitting alignment, but it achieves a suboptimal score among all fitting alignments.

Give an algorithm which computes the optimal fitting alignment. Explain how to fill in the first row and column of the dynamic programming table and give a recurrence to fill in the rest of the table. Give a method to find the best alignment once the table is filled in. The algorithm should run in time $O(nm)$.

Problem 6.31 (18 Points)

Given two strings \mathbf{v}_1 and \mathbf{v}_2 and a text \mathbf{w} , find whether there is an occurrence of \mathbf{v}_1 and \mathbf{v}_2 interwoven (without spaces) in \mathbf{w} . For example, the strings **abac** and **bbc** occur interwoven in **cabbabccdw**. Give an efficient algorithm for this problem.

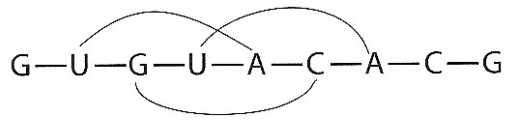
Problem 6.46 (18 Points)

Devise an algorithm to compute the number of distinct optimal local alignments (optimal paths in local alignment edit graph) between pairs of strings.

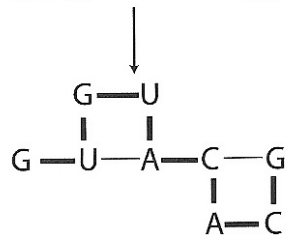
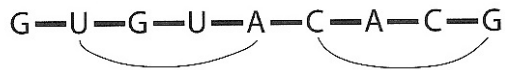
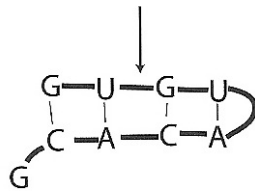
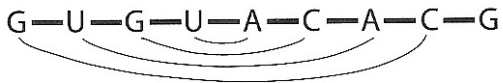
Problem 6.51 (18 Points)

RNAs adopt complex three-dimensional structures that are important for many biological functions. Pairs of positions in RNA with complementary nucleotides can form *bonds*. Bonds (i, j) and (i', j') are interleaving if $i < i' < j < j'$ and noninterleaving otherwise (fig. 6.30). Every set of noninterleaving bonds corresponds to a potential RNA structure. In a very naive formulation of the RNA folding problem, one tries to find a maximum set of noninterleaving bonds. The more adequate model, attempting to find a fold with the minimum energy, is much more difficult.

Develop a dynamic programming algorithm for finding the largest set of noninterleaving bonds given an RNA sequence.



(a) Interleaving bonds



(b) Non-interleaving bonds

Figure 6.30 Interleaving and noninterleaving bonds in RNA folding.