PROOF PAPER #1

NP-Completeness of Multiple Alignment with SP-Score

Paul Bodily

September 25, 2013

1 Introduction

Within a species' genome, biologists are primarily interested in the relatively small fraction of sequence that somehow contributes to the species' persistence. The vast majority of DNA in a genome does not serve any known purpose (i.e., it is "junk DNA"). One of the foremost methods for identifying functional sequence is to look for *conserved* sequence among several species. When a particular sequence serves an important function, changes in the sequence would render the organism inviable. This results in the change not being passed on to future generations. *Only progeny that keep these functionally-important sequences intact are likely to survive and reproduce* and thus these sequences tend to be *conserved* across species even after millions of years. For example, the Dyrk1a gene, which encodes for an important neurological protein of the same name and which has been implicated in Down syndrome, is highly conserved across several species (see Figure 1). It is important to be able to perform multiple sequence alignment in order to detect highly conserved subregions among a set of biological sequences.

A common metric for scoring a multiple sequence alignment is the *SP-score*, where SP stands for *sum of all pairs* [2]. A standard assumption about any score scheme s is that is satisfies *triangle inequality*, which is that for any three letters x, y, and z, $s(x,z) \leq s(x,y) + s(y,z)$.

Algorithms which optimally align multiple sequences under the SP measure are based on dynamic programming and require a running time that is in the order of the product of the lengths of the input strings [1]. The theorem being considered demonstrates that the decision version of the multiple sequence alignment problem is NP-complete [5].

2 Definitions

- A *sequence* is a string over some alphabet Σ. For DNA sequences the alphabet Σ contains four letters, namely *A, C, G,* and *T*, which represent four distinct nucleotides. For protein sequences, Σ contains 20 letters, each representing a unique amino acid.

- An *alignment* of two sequences s_1 and s_2 is obtained by inserting spaces into either sequence or at either end of either sequence such that the two resulting sequences s_1' and s_2' are of equal length. In other words every letter in s_1' is directly opposite to a letter in s_2'. Spaces, denoted here with

XP_002664656.1	51	FHAAGLQMAAPMPHSHQQYSDCHQQS-TDQSVTVLPYSDQTQALTAS---	96
NP_989881.1	23	FHAAGLQMAGQMSHSHQQYSDRRQQNINDQQVSASSYTDRIQQPLTN---	69
XP_849580.2	23	FHAAGLQMAGQMPHSH-QYSDRRQPNINDQQVSALSYSDQIQQPLTNQVM	71
NP_031916.1	23	FHAAGLQMAAQMPHSH-QYSDRRQPSISDQQVSALPYSDQIQQPLTNQVM	71
NP_036923.1	23	FHAAGLQMAAQMPHSH-QYSDRRQPNISDQQVSALSYSDQIQQPLTNQVM	71
NP_001192943.1	23	FHAAGLQMAGQMPHSH-QYSDRRQPNISDQQVSALSYSDQIQQPLTNQVM	71
XP_001083622.1	23	FHAAGLQMAGQMPHSH-PYSDRRQPSISDQQVSALAYSDQIQQPLTNQVM	71
NP_001387.2	23	FHAAGLQMAGQMPHSH-QYSDRRQPNISDQQVSALSYSDQIQQPLTNQVM	71
XP_514894.2	23	FHAAGLQMAGQMPHSH-QYSDRRQPNISDQQVSALSYSDQIQQPLTNQVM	71

XP_002664656.1	97	------QRHMPQCFRDPTLAPLRKLSIDLIKTYKHINEVYYAKKKRRHQQ	140
NP_989881.1	70	------QRRMPQTFRDPATAPLRKLSVDLIKTYKHINEVYYAKKKRRHQQ	113
XP_849580.2	72	PDIVMLQRRMPQTFRDPATAPLRKLSVDLIKTYKHINEVYYAKKKRRHQQ	121
NP_031916.1	72	PDIVMLQRRMPQTFRDPATAPLRKLSVDLIKTYKHINEVYYAKKKRRHQQ	121
NP_036923.1	72	PDIVMLQRRMPQTFRDPATAPLRKLSVDLIKTYKHINEVYYAKKKRRHQQ	121
NP_001192943.1	72	PDIVMLQRRMPQTFRDPATAPLRKLSVDLIKTYKHINEVYYAKKKRRHQQ	121
XP_001083622.1	72	PDIVMLQRRMPQTFRDPATAPLRKLSVDLIKTYKHINEVYYAKKKRRHQQ	121
NP_001387.2	72	PDIVMLQRRMPQTFRDPATAPLRKLSVDLIKTYKHINEVYYAKKKRRHQQ	121
XP_514894.2	72	PDIVMLQRRMPQTFRDPATAPLRKLSVDLIKTYKHINEVYYAKKKRRHQQ	121

XP_002664656.1	141	GQGEDSSHKKERKVFNDGYDDENYDYVVKNGEKWMDRYEIDSLIGKGSFG	190
NP_989881.1	114	GQGDDSSHKKERKVYNDGYDDDNYDYIVKNGEKWMDRYEIDSLIGKGSFG	163
XP_849580.2	122	GQGDDSSHKKERKVYNDGYDDDNYDYIVKNGEKWMDRYEIDSLIGKGSFG	171
NP_031916.1	122	GQGDDSSHKKERKVYNDGYDDDNYDYIVKNGEKWMDRYEIDSLIGKGSFG	171
NP_036923.1	122	GQGDDSSHKKERKVYNDGYDDDNYDYIVKNGEKWMDRYEIDSLIGKGSFG	171
NP_001192943.1	122	GQGDDSSHKKERKVYNDGYDDDNYDYIVKNGEKWMDRYEIDSLIGKGSFG	171
XP_001083622.1	122	GQGDDSSHKKERKVYNDGYDDDNYDYIVKNGEKWMDRYEIDSLIGKGSFG	171
NP_001387.2	122	GQGDDSSHKKERKVYNDGYDDDNYDYIVKNGEKWMDRYEIDSLIGKGSFG	171
XP_514894.2	122	GQGDDSSHKKERKVYNDGYDDDNYDYIVKNGEKWMDRYEIDSLIGKGSFG	171

Figure 1: A multiple sequence alignment of a subsection of the Dyrk1a protein in several species including human, dog, mouse, and zebra fish. The identifiers at the left each indicate a unique species and the aligned sequences are flanked by the start and end indices of the aligned region within the protein. The fact that a variety of species have each maintained a relatively conserved version of this protein suggests that it performs a very important function.

Δ, are also often called *gaps* and can be considered as either insertions into one sequence or a deletions from the other.

- A *match* refers to the case when a letter in s'_1 is opposite the *same* letter in s'_2. A *mismatch* is anything that is not a match or an insertion/deletion.
- An *alignment score* (also *value*) denotes the value of a particular alignment according to some *score scheme* s and is defined as $\Sigma_{i=1}^{l}\ s(s'_1(i), s'_2(i))$, where $s'_1(i)$ and $s'_2(i)$ denote the two letters at the ith column of the alignment, $s(s'_1(i), s'_2(i))$ is the score of the letters $s'_1(i)$ and $s'_2(i)$, and l is the length of the sequences $s'_1(i)$ and $s'_2(i)$.
- A *score scheme* is a mapping which maps each unique pair of letters in the sequence alphabet to an alignment score for the letters when aligned opposite each other in two sequences (e.g., Figure 2).

- An *optimal alignment* of two sequences is one that minimizes the alignment score over all possible alignments.
- The *edit distance* between two sequences is defined as the minimum alignment value of the two sequences.
- A *multiple alignment* A of $k \geq 2$ sequences is obtained as follows: spaces are inserted into each sequence so that the resulting sequences have the same length l, and the sequences are arrayed in k rows of l columns each. Again, a score value is defined on each column under some score scheme and the value of A is simply the sum of the scores of all columns.
- *SP-score* is a popular score scheme which defines the score value of a column as the sum of the scores of all pairs of letters in the column. The value of the alignment A thus corresponds to the sum of the values of all pairwise alignments induced by A.

3 Theorem Description

We now describe the proof of the NP-completeness of multiple sequence alignment with SP-score. One way to prove NP-completeness is to show that some other NP-complete problem can be reduced (i.e., transformed) into the new problem for which one would like to prove NP-completeness. The *shortest common supersequence* problem (presented by Garey and Johnson [3]) is NP-complete. We demonstrate its reduction to the problem of multiple alignment with SP-score.

The decision problem associated with the shortest common supersequence problem asks whether, given some set of sequences S and some integer m, there exists a sequence s whose length is less than or equal to m which is a supersequence of each sequence in S (in this case, a supersequence for a sequence t is defined as a sequence s which can be formed by prepending, inserting, or appending additional sequence to t). The problem remains NP-complete even if the alphabet for the sequences Σ contains as few as two letters [4], which is the case used to prove the reduction.

To complete the reduction, we must construct an instance of the supersequence problem that can be formulated as a multiple sequence alignment problem. To aid in constructing such an instance consider first this formulation of our multiple sequence alignment problem: Given a set S of sequences over alphabet {0,1}, and a positive integer m, we construct a collection of sets $X = \{X_{i,j} \mid i, j \leq 0, i + j = m\}$, where $X_{i,j} = S \cup \{a^i, b^j\}$ and a and b are two new letters. We assume that each sequence in S has length at most m. The score scheme is shown in Figure 2.

To show that multiple alignment with SP-score is NP-hard, it is sufficient to show that: S has a supersequence s of length m if and only if some $X_{i,j}$ has an alignment with value at most c. The reduction is essentially of the form:

$$P \iff Q$$

S	0	1	a	b	Δ
0	2	2	1	2	1
1	2	2	2	1	1
a	1	2	0	2	1
b	2	1	2	0	1
Δ	1	1	1	1	0

Figure 2: A score scheme used in the reduction of the shortest common supsersequence problem to the MSA with SP-score problem.

where

$$P = \text{“}S \text{ has a supersequence } s \text{ of length } m\text{”} \text{ and}$$
$$Q = \text{“}\exists X_{i,j} \in X(X_{i,j} \text{ has an alignment with value} \leq c)\text{”}$$

4 Logic-and-Proof Strategies Analysis

The proof is solved by first solving the reverse implication, $Q \implies P$ and then the forward implication, $P \implies Q$. We do this by finding appropriate values for m given c and then c given m. The author tells us up front the relationship between c and m which is that $c = (k - 1)|S| + (2k + 1)m$, where $|S|$ is the total length of all sequences in S. We will attempt to make sense of why this particular assignment for c was made as we explain the remainder of the proof.

4.1 Reverse Implication

In this step we prove that if $\exists X_{i,j} \in X(X_{i,j}$ has an alignment with value $\leq c$, then S has a supersequence s of length m. We assume it as given that we have an alignment A of the $k + 2$ sequences in $X_{i,j}$ with value at most c for some i, j. We now prove that S has a supersequence s of length m.

First, using Existential Instantiation, we must pick some $X_{i,j}$ for which the antecedent is true. To do this, consider the alignment of the original k sequences in S. Due to the score scheme, which always scores a match or mismatch as 2 and an insertion/deletion as 1, this alignment will always have a score of $(k - 1)|S|$. Now we begin to intuit the relationship between c and m. Because of our previously assigned value of c above, the total contribution of the pairwise alignments involving sequences a^i and/or b^j, is at most $(2k + 1)m$. Note that $2k + 1$ is the number of pairwise alignments involving a^i and/or b^j (2 with each of k sequences + 1 between a^i and b^j). Therefore the average value for each of these pairwise alignments must be less than or equal to m. As m is at least as long as each sequence in S, every 0 must be aligned with an a and every 1

must be aligned with a b in A. Otherwise, then due to the score scheme, the pairwise alignments involving a^i and/or b^j would include the addition of one or more 2's, causing the sum to exceed m. We can choose i and j such that there are enough a's and b's in a^i and b^j to construct such an alignment. This will be our selection for $X_{i,j}$.

Having thus more precisely defined the alignment A of the $k+2$ sequences in $X_{i,j}$ (which has a value at most c), we can obtain a supersequence s for S by assigning 0 to the columns in A containing a's and 1 to the other columns. The length of s is $i + j = m$.

4.2 Forward Implication

Having proven the reverse implication, we now prove that if S has a supersequence s of length m, then $\exists X_{i,j} \in X(X_{i,j}$ has an alignment with value $\leq c)$. We assume it as given that s is a supersequence for S with length m. Let i be the number of 0's and j be the number of 1's in s. We now show that $\exists X_{i,j} \in X(X_{i,j}$ has an alignment with value $\leq$ c).

We let $X_{i,j}$, as defined above, be arbitrary. For each sequence $t \in S$, there exists an alignment of t and s such that each 0 (or 1) in $X_{i,j}$ matches a 0 (or 1, respectively) in s. Some 0's and 1's in s may correspond to spaces. To obtain a multiple sequence alignment for S, we align each $t \in S$ with s in this manner and then align the a's in a^i with the 0's in s and the b's in b^j with the 1's in s. In this alignment, the letters in a column are either 0, a, Δ, or 1, b, Δ. The value of the alignment (with sequence s removed) is c.

4.3 *Quod erat demonstradum*

Therefore, by checking the value of an optimal alignment of $X_{i,j}$, $i + j = m$, we can use this value to answer if there is a supersequence s for X with length m using a polynomial-time reduction. Because we know that the latter is an NP-complete problem, we know that the multiple sequence alignment problem with SP-score is also NP-complete.

5 Example

Consider a set S = 0100, 0010. As shown in Figure 3, their optimal alignment has a value of 8.

We construct a collection of sets $X = \{X_{i,j} \mid i, j \leq 0, i + j = m\}$, where $X_{i,j} = S \cup \{a^i, b^j\}$ and a and b are two new letters. m must be greater than or equal to the largest sequence in S in length. We will choose m=5. X is shown in Figure 4. The optimal pairwise alignments for $X_{4,1}$ involving a^4 and b^1 is shown in Figure 5. The value c is calculated as the sum of the alignments in Figure 5, i.e., $c = 33$ (we leave it to the reader to prove that the score for the alignment is indeed 33).

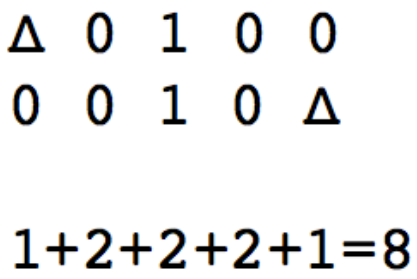

Figure 3: An optimal alignment for the two sequences shown is given together with the computed alignment score.

```
X={X0,5={"0100","0010","","bbbbb"},
X1,4={"0100","0010","a","bbbb"},
X2,3={"0100","0010","aa","bbb"},
X3,2={"0100","0010","aaa","bb"},
X4,1={"0100","0010","aaaa","b"},
X5,0={"0100","0010","aaaaa",""}}
```

Figure 4: X is a collection of sets defined from a set of sequences and facilitates the reduction from the shortest common superstring problem.

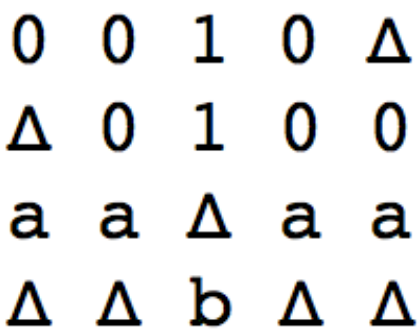

Figure 5: The optimal pairwise alignment for $X_{4,1}$ involving a^4 and b^1.

We know then that $\exists X_{i,j} \in X(X_{i,j}$ has an alignment with value $\leq c)$ and we can obtain a supersequence s for S of length m where m is calculated as follows:

$$c = (k \text{ - } 1)|S| + (2k + 1)m$$
$$33 = (2 \text{ - } 1)8 + (2(2) + 1)m$$
$$5 = m$$

We do this by assigning 0 to the columns in A containing a's and 1 to the

other columns (see Figure 6).

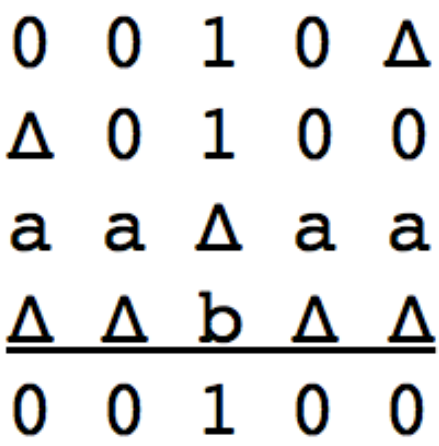

Figure 6: A derivation of the supersequence s for S of length $m = 5$.

Conversely, had we started with this supersequence s for S of length $m = 5$, we would be able to find an alignment for some $X_{i,j} \in X$ (which is derived from S) such that $X_{i,j}$ has an alignment with value $\leq c$, where c is calculated as follows:

$$c = (k - 1)|S| + (2k + 1)m$$
$$c = (2 - 1)8 + (2(2) + 1)5$$
$$c = 33$$

This alignment is found by finding an alignment for each sequence $t \in S$ and s such that each 0 (or 1) in $X_{i,j}$ matches a 0 (or 1, respectively) in s. We then align the a's in the sequence a^i with the 0's in s and the b's in b^j with the 1's in s. This is essentially the reverse process illustrated earlier in Figure 6.

In our example, by checking the value of an optimal alignment of $X_{i,j}$ where $i + j = 4 + 1 = 5 = m$, we are able to answer if there exists a supersequence s for X with length m.

References

[1] Stephen F Altschul and David J Lipman. Trees, stars, and multiple biological sequence alignment. *SIAM Journal on Applied Mathematics*, 49(1):197–209, 1989.

[2] David J Bacon and Wayne F Anderson. Multiple sequence alignment. *Journal of molecular biology*, 191(2):153–161, 1986.

[3] Michael R Garey and David S Johnson. Computers and intractibility, 1979.

[4] Martin Middendorf. More on the complexity of common superstring and supersequence problems. *Theoretical Computer Science*, 125(2):205–228, 1994.

[5] Lusheng Wang and Tao Jiang. On the complexity of multiple sequence alignment. *Journal of computational biology*, 1(4):337–348, 1994.