Sharper Upper and Lower Bounds for Consensus-Pattern

Broňa Brejová Daniel G. Brown Ian M. Harrower Alejandro López-Ortiz
Tomáš Vinař

School of Computer Science, University of Waterloo

Abstract

We present sharper upper and lower bounds for a known polynomial approximation scheme due to Li, Ma and Wang [5] for the Consensus-Pattern problem. This NP-hard problem is an abstraction of a common bioinformatic discovery task, and numerous heuristic programs exist to solve it in practice. The PTAS due to Li *et al.* is simple, and a preliminary implementation [6] gave reasonable results in practice. However, the previously known bounds on its performance are useless when runtimes are actually manageable. Here, we present lower and upper bounds on the performance of this algorithm that partially explain why its behavior is better in practice than what has been predicted in theory, and which still give specific examples of instances of the problem for which the PTAS performs poorly in practice.

1 Introduction

Bioinformaticists often find themselves with several different DNA or protein sequences that are known to share a particular function, but where the origin in the sequence of the function is unknown. For example, suppose one has the DNA sequence of the region surrounding several genes, known to be regulated by a particular transcription factor. Here, the shared regulation may be caused by a sequence element common to all, to which the transcription factor binds. Discovering this experimentally is very expensive, so computational approaches can be helpful to limit searches.

The motif discovery problem is an abstraction of this problem. In it, we are given n sequences, all of length m, over an alphabet Σ. We seek a single motif, of length L that is found approximately as a substring of all sequences. Several variants of this problem exist. One can seek to minimize the maximum Hamming distance between the motif and its instances in all strings (*e.g.* [8, 2]), maximize the information content (minimize the entropy) of the chosen motif instances (*e.g.* [3, 1, 4]), or minimize the total of the Hamming distances between the motif and its instances [5]. This latter problem can be formally defined as follows:

Definition 1 (Consensus-Pattern). *Given are n sequences $s_1, \ldots, s_n$ of length m each over an alphabet of size A. Find a substring t_i of given length L in each of the sequences and a median string s of length L so that the total Hamming distance $\sum_i d_H(s, t_i)$ is minimized.*

Li, Ma and Wang [5] give a very simple polynomial-time approximation scheme (PTAS) for this combinatorial motif problem. For a given value of r, consider all choices of r substrings of length L from the n sequences (the same substring may occur multiple times). For each such collection $\mathcal{C}$ of substrings, we compute its consensus by identifying the most common letter in the first position of each substring, the second position, and so on, producing a motif $M_{\mathcal{C}}$. It is easy to identify for a

Condition	New results		Previous upper bound
	Lower bound	Upper bound	
$r = 1$	2	2	N/A
$r = 3$	1.5	≈ 1.528	$\approx 1 + 4.006 \cdot (A - 1)$
general r binary alphabet	$1 + \Theta(1/r^2)$ conjecture: $1 + \Theta(1/\sqrt{r})$		$1 + \Theta(1/\sqrt{r})$
general			$1 + \Theta(A/\sqrt{r})$

Table 1: **Overview of the results.**

given motif $M_{\mathcal{C}}$ its closest match in each of the n sequences, and thus its score. We do this for all $n^r(m - L + 1)^r$ collections of substrings, and pick the collection with the best score. The algorithm has $O(L(nm)^{r+1})$ run time, and is thus polynomial-time for any particular value of r. Li *et al.* also give a quite large bound on the worst-case approximation ratio of this algorithm for $r \geq 3$:

$$1 + \frac{4A - 4}{\sqrt{e}\left(\sqrt{4r + 1} - 3\right)}, \tag{1}$$

where A is the size of the alphabet Σ. For example, for $r = 3$, this gives us $O(L(nm)^4)$ algorithm with approximation ratio of ≈ 13 even for DNA sequences. To achieve reasonable approximation ratio of 2, we would have to use $r \geq 8$ for DNA sequences ($A = 4$), or $r \geq 27$ for protein sequences ($A = 20$), thus requiring unreasonable running times. The high value of the proven bound would seem to suggest that the algorithm will be useless in practice.

However, many successful combinatorial motif finders do work by generalizing from small samples in this way, such as SP-STAR [**?**] and CONSENSUS (samples of 1) [3], COMBINE (samples of 2 to 3) [7], COPIA (samples of arbitrary size) [6]. Here, focusing on Li *et al.*'s PTAS, we show tighter bounds on its performance much closer to reasonable numbers for practical values of r. We also provide first lower bounds on the PTAS's performance, identifying specific examples of the problem for which the algorithm performs poorly. Our results are summarized in Table 1.

2 Basic Observations

Assume that the PTAS achieves some approximation ratio α for a given r, L, and sequences $s_1, \ldots, s_n$. Now consider only the true occurrences $t_1, \ldots, t_n$ of the optimal motif s^*. If we run the same PTAS on input strings $t_1, \ldots, t_n$, we would get solution with approximation ratio at least α, because we consider fewer collections of r substrings $\mathcal{C}$, and for each such collection we have also fewer choices to find a good occurrence of the consensus string $M_{\mathcal{C}}$. Therefore we need to consider only inputs in which $m = L$, i.e. all input strings have the same length as the desired motif.

Note that if $m = L$, the problem can be trivially solved by finding the consensus string of all input sequences. However, we may still apply the PTAS described above, which will in this special case work as follows:

1. choose parameter $r \geq 1$
2. for every collection $\mathcal{C}$ of r strings from the set $\{s_1, \ldots, s_n\}$
   — set $M_{\mathcal{C}}$ to be the consensus string of string in $\mathcal{C}$
   — compute cost $\sum_i d_H(M_{\mathcal{C}}, s_i)$

3. choose the best $M_{\mathcal{C}}$ as the median string (motif)

To simplify the notation, we will assume that the alphabet is $\{0, 1, \ldots, A-1\}$. In the special case $m = L$, we will also always renumber the characters in each column so that consensus is 0 and therefore the overall optimal motif is $s = 0^L$ (this cannot be done in general if $m > L$).

Finally, we can encounter the problem of ties, that is, a situation when consensus string u of some collection $\mathcal{C}$ is not unique. Consider for example $r = 3$ and input strings `01`, `02`, `10`, `20`. The optimal motif is `00` with cost 4. If $\mathcal{C}$ contains the first three strings, the consensus $M_{\mathcal{C}}$ can be any of the strings `00`, `01`, and `02`. The first of them is optimal, but the latter two have cost 5.

It is not realistic to assume that the PTAS will always choose the best out of all possible consensus strings, because their number can be exponential in L. For simplicity, we will assume that the PTAS will choose the worst consensus u out of all possibilities (i.e., in the above example it would choose `01` or `02`).

3 Upper Bounds

In this section we give better upper bounds for practical values of $r = 1$ and $r = 3$.

Theorem 1. *The approximation ratio of the PTAS is at most 2, even for $r = 1$ and regardless of the size of the alphabet.*

Proof. Let c be the cost of the optimal motif 0^L, that is, the total number of non-zero elements in all sequences. Let a_i be the number of non-zero elements in sequence s_i. If the PTAS chooses sequence s_i as the motif, the cost will increase by at most n for every column where s_i has non-zero element. Therefore the cost will be at most $c + na_i$. When we sum this quantity over all sequences s_i, we get $nc + n\sum_{i=1}^{n} a_i = 2nc$. Since the sum of costs for n different potential motifs is at most $2nc$, at least one of the motifs has cost at most $2c$, which means the approximation ratio is at most 2. □

Theorem 2. *The approximation ratio of the PTAS for $r = 3$ is at most $(64 + 7\sqrt{7})/54 \approx 1.528$ regardless of the size of the alphabet.*

Proof. Let p be proportion of zeroes and $q = (1-p)$ be proportion of non-zeroes in input sequences. The optimal cost is therefore qnL. Let b_j be the number of non-zeroes in column j.

For each column, we can estimate the expected cost of the column if the triple of rows is chosen uniformly at random. Column with b non-zeroes will get non-zero answer only if two or three rows in the triple are non-zeroes. There are $b^3 + 3b^2(n-b)$ such triples. Each of these triples will incur cost of at most n in this column. The consensus will be zero for triples with two or three zeroes (their number is $(n-b)^3 + 3(n-b)^2 b$). Each of these triples will incur cost b in this column.

Thus the expected cost $E(b)$ for a column with b non-zeroes is at most $C(b)/n^3$, where $C(b)$ is the sum of costs over all triples of rows:

$$C(b) = b^3 n + 3b^2(n-b)n + (n-b)^3 b + 3(n-b)^2 b^2 = bn^3 + 3b^2 n^2 - 5b^3 n + 2b^4. \tag{2}$$

From linearity of expectation, expected cost over all columns is

$$E(b_1, \ldots, b_L) = \sum_{j=1}^{L} E(b_j) = \frac{1}{n^3} \cdot \sum_{j=1}^{L} C(b_j). \tag{3}$$

There must exist a triple, whose cost is at most $E(b_1, \ldots, b_L)$. Such triple achieves approximation ratio $E(b_1, \ldots, b_L)/nqL$.

We will prove by induction on L that $E(b_1, \ldots, b_L) \leq HnqL$, where $H = (64 + 7\sqrt{7})/54$. This implies that H is an upper bound on approximation ratio.

For $L = 1$, the approximation ratio is

$$E(qn)/nqL = 2q^3 - 5q^2 + 3q + 1$$

with maximum reached for $q = \frac{5-\sqrt{7}}{6}$ equal to H.

Now assume that the induction hypothesis is true for $L - 1$. We will prove that it is also true for L. Expected cost of the first column is $E(b_1)$. By induction hypothesis, the expected cost of the remaining $L - 1$ columns is at most $(nqL - b_1) \cdot H$ ($nqL - b_1$ is the optimal cost for the remaining columns). Therefore:

$$E(b_1, \ldots, b_L) \leq E(b_1) + (nqL - b_1) \cdot H = \underbrace{\frac{2b^4 - 5b^3 n + 3b^2 n^2 + (1-H)bn^3}{n^3}}_{(*)} + HnqL \qquad (4)$$

We want to prove, that (*) is always negative for $0 \leq b \leq n$. Indeed, (*) can be simplified as $(b/(108n^3)) \cdot (6b - (5+2\sqrt{7})n) \cdot (6b - (5-\sqrt{7})n)^2$. The first and third factors are always non-negative, and the second term cannot be positive unless $b > n$. Therefore the whole term is negative on our interval. □

It is, in fact, possible to easily characterize the "worst-case" scenario that maximizes $E(b_1, \ldots, b_L)$: this is achieved when the non-zero elements are distributed almost equally among the columns as follows.

Lemma 1. *For a given q, n, and L, $E(b_1, \ldots, b_L)$ is maximized, when for some $k \leq L$, $b_1, \ldots, b_k = 0$, and $b_{k+1} = b_{k+2} = \ldots = b_L \leq n$ (if we allow $b_1, \ldots, b_L$ to be non-integral).*

Proof. (by induction on L). For $L = 1$, the hypothesis holds trivially.

Let us assume that the hypothesis holds for all $L' < L$. Without loss of generality, we can assume that the columns are sorted by b_j. If $b_1 = 0$, the hypothesis holds trivially from induction hypothesis. Let $b_1 > 0$. Then, by induction hypothesis, all the rest of the columns must by distributed equally (there are no columns with $b_i = 0$, since b_1 is the smallest). The cost will be therefore:

$$C(b_1) + (L-1) \cdot C\left(\frac{Lnq - b_1}{L-1}\right), \qquad (5)$$

where $nL(q-1) + n \leq b_1 \leq qn$, and $b_1 > 0$. This is indeed maximized for $b_1 = qn$ (straightforward, but technical proof not shown). □

4 Lower Bounds

In this section we present first lower bounds for the Li et al.'s PTAS. For small values of r, we are able to give lower bounds which almost match our upper bounds from previous section. For general values of r, we show an example where the PTAS has approximation ratio $1 + \Theta(1/r^2)$. Finally, we conjecture that lower bound on approximation ratio matches asymptotically the upper bound $1 + \Theta(1/\sqrt{r})$; to support this claim, we present an example for which a slightly modified algorithm has approximation ratio $1 + \Theta(1/\sqrt{r})$.

Theorem 3. *For $r = 1$, the approximation ratio is at least 2, even for binary alphabet.*

Proof. We set $L = n$. The input will be the unit matrix of size $n \times n$ with ones on the diagonal and zeroes everywhere else. The cost of the optimal solution is n. The result of the PTAS for $r = 1$ will be one of the matrix rows, with cost $2n - 2$. The approximation ratio is therefore $2 - 2/n$ which goes to 2 as n goes to infinity. This shows that the upper bound 2 is tight for $r = 1$. □

Theorem 4. *For $r = 3$, the approximation ratio is at least 3/2.*

Proof. Consider the following example:

```
0   1
.   .
.   .
.   .
0   k
1   0
.   .
.   .
.   .
k   0
```

The optimal solution is 00 with cost $2k$. However, for any three strings, the solution will be $0x$ or $x0$, which has cost $3k - 1$. □

Theorem 5. *The approximation ratio of the PTAS is at least 1+$\Theta(1/r^2)$.*

Proof. We create $n = r + 2$ sequences, each of length $L = (r + 5)/2$. The first $L - 1$ columns of the first $L - 1$ sequences will be an inverted identity matrix, with zeroes on the diagonal and ones everywhere else. The last column of these sequences contains zeroes. The remaining $n - L + 1$ sequences have zeroes in the first $L - 1$ columns and one in the last column. For example for $r = 5$ we get the following input:

```
0 1 1 1 0
1 0 1 1 0
1 1 0 1 0
1 1 1 0 0
0 0 0 0 1
0 0 0 0 1
0 0 0 0 1
```

Assume that the PTAS can obtain the optimal solution 0^L. Then there must be some collection $\mathcal{C}$ of strings such that each column has more than $r/2$ zeroes. In particular, for the last column, more than half of these strings are chosen from the first $L - 1$ sequences of the input. Thus, to achieve more than $r/2$ zeroes in any other column $i < L$, we have to include at least one copy of sequence i (less than $r/2$ copies of the last $n - L + 1$ sequences are included). That means we need to include each of the first $L - 1$ sequences, and therefore each of the first $L - 1$ columns contain at least $L - 2 = (r + 1)/2$ ones. This is a contradiction. Therefore PTAS cannot achieve the optimal solution.

The optimal solution in the above example has cost $c = (r + 1)(r + 5)/2$. The PTAS will find motif $0^{L-1}1$ with cost $c+1$. Therefore the approximation ratio is $1+1/c = 1+4/[(r+3)^2-4]$. □

Theorem 6. *Consider a modified PTAS, where we allow only a single sample from each input sequence. Such modified algorithm has approximation ratio at least* $1 + \Theta(1/\sqrt{r})$*, even for binary alphabet.*

Proof. Consider the following example. For a given odd r, the number of sequences is $n = 2r$. Every column has $o = r - \sqrt{r}$ ones and $n - o = r + \sqrt{r}$ zeroes. There is one column for each possible combination of o ones and $n - o$ zeroes. This means we have $L = \binom{n}{o}$ columns. In this instance the optimal cost is $c = \binom{n}{o} o$.

Note, that when choosing r sequences to form a motif, it does not matter which r sequences we choose; all options are symmetric up to reordering of columns. Therefore all choices lead to the same cost and we can assume without loss of generality, we have selected first r rows.

For every k, denote $K(k)$ number of columns that have exactly k ones in the first r rows. This can be computed simply as follows:

$$K(k) = \binom{r}{k} \cdot \binom{r}{r - \sqrt{r} - k} = \binom{r}{k} \cdot \binom{r}{k + \sqrt{r}} \tag{6}$$

Let K be the number of columns that yield 1 in the consensus pattern. Each of these columns contains at least $(r+1)/2$ ones in the first r rows. Therefore, for $r \geq 25$:

$$K = \sum_{k=\frac{r+1}{2}}^{r-\sqrt{r}} K(k) \geq \sum_{k=\frac{r+1}{2}}^{\frac{r+1}{2}+\sqrt{r}} K(k) \geq \sum_{k=\frac{r+1}{2}}^{\frac{r+1}{2}+\sqrt{r}} \binom{r}{k} \cdot \binom{r}{k + \sqrt{r}} \geq \sqrt{r} \cdot \binom{r}{\frac{r+1}{2} + 2\sqrt{r}}^2 \tag{7}$$

According to Maple:

$$\lim_{r\to\infty} \frac{K}{L} \geq \lim_{r\to\infty} \frac{\sqrt{r} \cdot \binom{r}{\frac{r+1}{2} + 2\sqrt{r}}^2}{\binom{2r}{r + \sqrt{r}}} = \frac{2e^{-15}}{\sqrt{\pi}} = a \tag{8}$$

We have shown that regardless of choice of rows, there are at least aL columns with consensus 1 instead of 0, as r approaches infinity. Each of these columns will contribute $2\sqrt{r}$ more than the optimal solutions. Therefore, the cost of the solution given by PTAS is at least $c + 2aL\sqrt{r}$, and therefore the approximation ratio is at least $c/(c + 2aL\sqrt{r}) = 1 + \frac{2a\sqrt{r}}{r+\sqrt{r}} = 1 + \Theta(1/\sqrt{r})$. □

References

[1] T.L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proceedings of the 2nd International Conference on Intelligent Systems for Molecular Biology (ISMB 1994)*, pages 28–36. AAAI Press, 1994.

[2] J. Buhler and M. Tompa. Finding motifs using random projections. In *Proceedings of the 5th Annual International Conference on Computational Molecular Biology (RECOMB 2001)*, pages 69–76, 2001.

[3] G.Z. Hertz and G.D. Stormo. Identifying dna and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics*, 15(7-8):563–577, 1999.

[4] C.E. Lawrence, S.F. Altschul, M.S. Boguski, J.S. Liu, A.F. Neuwald, and J.C. Wootton. Detecting subtle sequence signals: a Gibb's sampling strategy for multiple alignment. *Science*, 262(5131):208–214, 1993.

[5] M. Li, B. Ma, and L. Wang. Finding similar regions in many strings. *Journal of Computer and System Sciences*, 65(1):73–96, 2002.

[6] Chengzhi Liang. COPIA: A New Software for Finding Consensus Patterns in Unaligned Protein Sequences. Master's thesis, University of Waterloo, October 2001.

[7] Jiang Liu. A Combinatorial Approach for Motif Discovery in Unaligned DNA Sequences. Master's thesis, University of Waterloo, March 2004.

[8] P.A. Pevzner and S. Sze. Combinatorial approaches to finding subtle signals in dna sequences. In *Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology (ISMB 2000)*, pages 269–278, 2000.