IMPROVING COMPUTATIONAL EFFICIENCY FOR RNA SECONDARY STRUCTURE PREDICTION VIA DATA-ADAPTIVE ALIGNMENT CONSTRAINTS.

Angela D'Orazio[1] *and Gaurav Sharma*[1,2]

[1]Department of Electrical and Computer Engineering, University of Rochester,
Hopeman 204, RC Box 270126, Rochester NY 14627, USA
[2]Department of Biostatistics and Computational Biology, University of Rochester Medical Center,
601 Elmwood Avenue, Box 630, Rochester, NY 14624, USA

annjellah@gmail.com, gaurav.sharma@rochester.edu

ABSTRACT

The most accurate methods for RNA secondary structure prediction simultaneously predict the common structure and alignment among multiple homologs. In addition to dynamic programming, practical algorithms utilize heuristics to restrict the search space and further reduce time and memory requirements. This work is directed toward improving these heuristics in order to reduce computation without a compromise in accuracy. In this paper, a new, principled method for restricting the alignment search space in Dynalign [1] is introduced. Our results indicate that we are able to improve runtime with little affect on the accuracy of the structure predictions. This work utilizes Dynalign, but this method is also applicable to other structure prediction programs.

1. INTRODUCTION

Methods are desired to perform systematic searches over genomes in search of undiscovered, functional RNA. Knowledge of non-coding RNA (ncRNA) secondary structure provides information to predict its function. As a result, much research has been devoted to developing techniques to predict secondary structure from RNA sequence data. Reducing computational complexity and improving the accuracy of these programs are major hurdles that still need to be overcome.

From an evolutionary standpoint, preservation of structure in functional RNA is more important than preservation of the primary sequence [3] since homologs may maintain the same structure but have very low sequence similarity. Tertiary structure can often be inferred from secondary structure, thus, current approaches in research focus on RNA secondary structure prediction.

Techniques that predict secondary structure for single input sequences [3, 4] suffer from relatively low accuracy compared with comparative analysis methods, which utilize multiple homologs. However, accurate comparative analysis predictions require thousands of known homologs and vast amounts of time and manpower [5]. It is of interest to be able to predict secondary structure with a minimal number of known RNA homologs, with the goal of minimizing the prior knowledge and manpower needed while improving performance over single-sequence predictions. The most accurate of these approaches simultaneously align and fold a set of RNA sequences. Most algorithms that use this approach are based on the Sankoff algorithm: a dynamic programming algorithm which simultaneously explores all possible alignments and all possible base pairings of RNA sequences [6]. This algorithm is computationally demanding so current practical implementations [1, 7, 8, 9] employ heuristics to reduce time and memory requirements. Dynalign [1] employs a free energy minimization approach to this problem and is among the most accurate of these algorithms.

Dynalign simultaneously predicts an alignment and common secondary structure for a pair of sequence by optimizing a cost function [1]. Optimization of the cost function for all possible alignments and common secondary structures is computationally intractable on current hardware, so both the structure and alignment search spaces are restricted to improve efficiency.

This work improves the computational complexity of Dynalign by utilizing a novel method to restrict the alignment search space. This method builds on work by Harmanci et al [2], by using an adaptive approach to choosing probability thresholds. The new alignment space restriction is constructed so that in regions where there is high confidence in the maximum a posterior probability (MAP) alignment, we can restrict this space more, and in regions where there is low confidence the search space is expanded, relative to using a constant threshold for the entire alignment space. Our results demonstrate that it indeed provides improved execution time with little or no compromise in accuracy.

Section 2 describes the previous work employed to restrict the alignment space in Dynalign. Our novel method is presented in Section 3, and Section 4 shows the results obtained with our method and those of previous work. Section 5 presents future work that can be done to improve the alignment space restriction.

2. ALIGNMENT SPACE RESTRICTION IN DYNALIGN

Restriction of the alignment space was originally proposed by Sankoff [6], and was implemented in Dynalign

1-4244-2372-9/08/$20.00 ©2008 IEEE

[1] to make the computation feasible without sacrificing significant accuracy. In this method, the search space is restricted to a banded region by what is known as an "M-parameter," and is motivated by the idea of imposing a maximum allowable insertion length. This method is cumbersome since it must be entered manually and requires knowledge of maximum insertion lengths for different RNA families.

Motivated by the desire for a data adaptive constraint, a modification by Harmanci et al. restricts the search space using a probabilistic approach. This is accomplished by excluding improbable alignment positions [2]. A hidden Markov model (HMM) is used to model pair-wise sequence alignment, as well as provide estimates of the a-posterior probabilities (APPs) of nucleotide *co-incidence* [2] with relatively low computational complexity. The search space is restricted by thresholding the APPs of nucleotide co-incidence, which is a probability surface as shown in Figure 1. The restricted search space allows only alignment positions whose APPs exceed a constant threshold. The optimal threshold varies depending on the similarity between the two RNA sequences. This is more principled than the ad hoc M-parameter, which arbitrarily restricted the maximum distance between two nucleotides that can be aligned. Using this method, Harmanci et al were able to improve runtime as well as eliminate manual parameter selection.

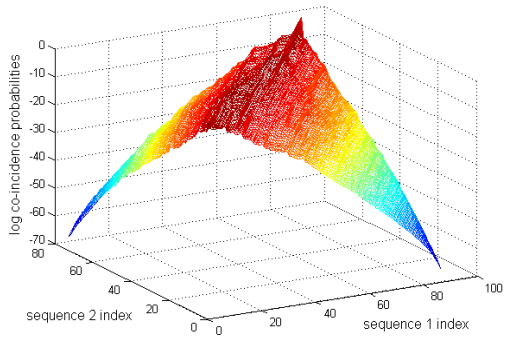

Figure 1: Output of the HMM: Probability surface illustrating log of posterior co-incidence probabilities

The speed of this algorithm is still problematic, however, especially considering the time to search an entire genome for novel ncRNA. For a test set used by Babak et al. to systematically evaluate ncRNA search tools, they determined that Dynalign would have used a year of CPU time, and would have taken 50-500 times longer than the other algorithms which they evaluated [10]. It is obvious that additional improvement in runtime is desirable.

3. NOVEL PROBABILISTIC ALIGNMENT CONSTRAINT FOR JOINT SECONDARY STRUCTURE PREDICTION

Our goal was to find a new way restrict the alignment search space to reduce runtime without sacrificing accuracy. Our novel method [12] attempts to ensure that the alignment envelope will contain the maximum a-posterior probability (MAP) co-incidence path as well as a region surrounding this path to exploring alternate alignments.

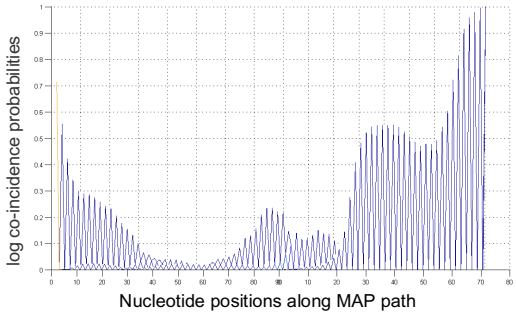

Figure 2: Illustration of probability surface along the MAP path.

In areas where there is good correspondence between the two sequences, the APPs corresponding to the MAP path are significantly higher. This can be seen in Figure 2, which shows a side view of the APPs from Figure 1 on a linear scale. From this we determined that in regions of high confidence in the MAP path, we can restrict the search space more. In regions of the MAP path where the sequences have insertions or substitutions there will be more uncertainty in the MAP path, so the corresponding APPs will be lower. In these regions we want to increase the search space.

The difference between the Harmanci et al. fixed threshold method and our novel adaptive threshold method is schematically illustrated in Figure 3. The diagram shows that by using an adaptive threshold, we can restrict the search space in areas of high confidence more than with the fixed threshold.

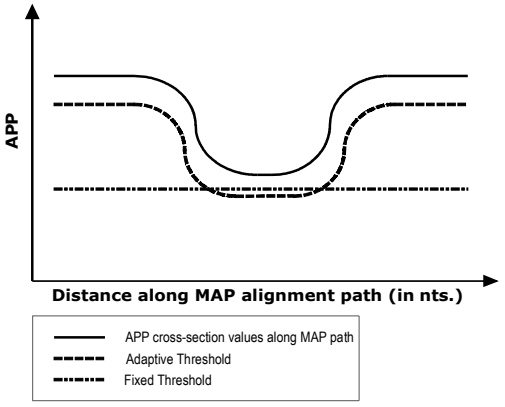

Figure 3: Comparison of adaptive threshold and fixed threshold: Cross-section of APPs observed along MAP path is shown.

Also, as illustrated by Figure 3**,** it may be possible to improve accuracy, by increasing the search space in areas of low confidence. To accomplish this, we use both the a-posterior probabilities (APP) of co-incidence from the HMM, as well as the output from the MAP alignment. These are already computed in the work by Harmanci et al [2] which minimizes additional computation in our work.

3.1. Determination of adaptive thresholds

A matrix of thresholds is used, denoted **Thresh**, with each element of the **Thresh** matrix corresponding to an element of the matrix of APPs. The $(i,k)^{th}$ element of the APP matrix, APP(i, k), denotes the *a posteriori* probability that i k, given the two RNA sequences. The notation i k indicates that the nucleotide at position i in the first sequence is co-incident [2] with the nucleotide at position k in the second sequence. Figure 1 illustrated the APPs for a sample sequence pair. Also, the MAP path, denoted **M,** is the set of all nucleotide positions that are co-incident in the MAP alignment path, and is formally defined as follows:

$$\mathbf{M}=\{(n_1,n_2): n_1 \quad n_2 \text{ in the MAP path}\} \qquad (1)$$

The value of **Thresh**(i, k) is set proportional to the minimum APP value over all elements of **M** which also fall in the corresponding row (i) and column (k) of the current threshold position. Mathematically, we denote an intermediate 2-D array **T**(i, k) as follows:

$$\mathbf{T}(i,k)=\min\left(\begin{array}{l}\min\limits_{i':(i',k)\ \mathbf{M}}\left(\mathbf{APP}(i',k)\right),\\ \min\limits_{k':(i,k')\ \mathbf{M}}\left(\mathbf{APP}(i,k')\right)\end{array}\right) \qquad (2)$$

where **T** is a matrix of probabilities, and each element, **T**(i,k) is the minimum of all APP values that lie in the corresponding row (i) or column (k) which also lies in the MAP co-incidence path. From **T**, the threshold values which will restrict the alignment search space are easily determined:

$$\mathbf{Thresh}(i,k)= \quad \mathbf{T}(i,k) \qquad (3)$$

where α is a constant < 1, which sets the threshold to be a fraction of the minimum APP in the MAP path. The minimum value is used in (2) above because when (i, k) lies in an insertion run, this represents a conservative threshold, favoring inclusion of more alignment positions. When the APP is high in the MAP path, this implies confidence in the alignment, and we can set the thresholds near this point are set higher to narrow the search space. If the APP happens to be relatively low in the MAP path, then we have a lower confidence in the alignment, and our method reduces the threshold near this area.

3.2. Determination of optimal threshold parameter based on sequence similarity

The alignment search space required is highly dependent on sequence similarity, which was verified when examining preliminary results using a single parameter value for all sequence similarities. It was determined that the optimal α-parameter should be chosen based on percentage sequence similarity. This does not add significant complexity, since the sequence similarity is already computed and used to select appropriate HMM parameters [2].

To choose the parameter, α, we can use consensus alignments from the RFAM database [11] and determine the value of alpha is necessary for each true alignment position to be included in the alignment envelop. Mathematically, it is noted that the following constraint must be met for a true alignment located at position (i, k) to be included in the search space:

$$\mathbf{T}(i,k) \quad \mathbf{APP}(i,k) \qquad (4)$$

It is then easy to determine the value of α required:

$$\rightarrow = \frac{\mathbf{APP}(i,k)}{\mathbf{T}(i,k)} \qquad (5)$$

The optimal parameter was determined experimentally using a test set of 3000 random RNA pairs with known pair-wise alignments. The known alignments are from the tRNA and 5sRNA families, and pairs were randomly selected from the RFAM database seed alignments. The sequences were categorized into bins by their percentage sequence identity, and then maximum values of alpha are computed for each. This data was used to determine the empirical probability of excluding a true alignment position as a function of alpha, and is shown in Figure 4. This is very similar to the procedure in [2] for determining the appropriate probability threshold.

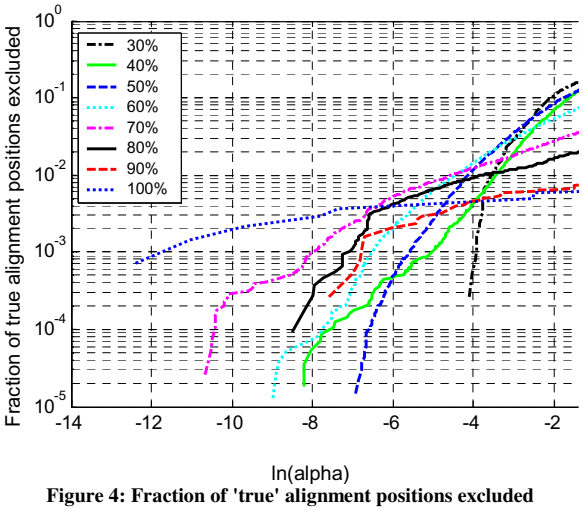

Figure 4: Fraction of 'true' alignment positions excluded as a function of the parameter α.

Figure 4 was used to choose appropriate values of α. For each sequence identity, the parameter alpha was chosen so that the empirical probability of excluding a true alignment position was less than 10^{-3}. From experimentation, choosing a probability lower than this would result in runtimes that were too slow. The appropriate parameter for each bin was chosen and incorporated into Dynalign.

4. RESULTS

200 tRNA and 200 5sRNA pair-wise alignments were used in each of the experiments performed. The consen-

sus secondary structure for each of the inputs was derived from hand-curated techniques and reported in the RFAM database [11].

4.1. Accuracy of Probabilistic Constraints

The accuracy of secondary structure prediction is evaluated in terms of Sensitivity and Positive Predictive Value (PPV). Sensitivity is the fraction of base pairings in the consensus structure that are correctly predicted in by Dynalign. PPV is the fraction of base pairings in the predicted structure that were correctly predicted. For both measures, slippage of a single nucleotide [2] is still considered to be a correct prediction since getting the correct topology is more important than the exact base pairings.

4.2. Performance of Adaptive Thresholding

Dynalign was implemented so that the parameter, α, varies based on the percentage sequence identity of the input sequences, as described in Section 3.2. We compare the performance to the fixed threshold method used in [2]. The results for tRNA and 5sRNA are shown in Table 1 and Table 2 respectively.

Table 1: Results for tRNA

		Fixed threshold	Adaptive threshold
PPV	Ave	0.94	0.95
	Min	0.26	0.24
	Max	1	1
Sensitivity	Ave	0.93	0.93
	Min	0.30	0.24
	Max	1	1
Runtime (s)	Ave	7.04	5.44
	Min	0.59	0.54
	Max	43.76	42.9

Table 2: Results for 5sRNA

		Fixed threshold	Adaptive threshold
PPV	Ave	0.81	0.81
	Min	0.20	0.20
	Max	0.97	0.97
Sensitivity	Ave	0.87	0.86
	Min	0.22	0.22
	Max	1	1
Runtime (s)	Ave	33.99	29.38
	Min	1.93	1.15
	Max	259.41	396.47

These results are significantly better than the previous method. For tRNA's we see a 23% reduction in runtime with no sacrifice in accuracy. There is even a slight improvement in PPV over the previous method. For 5sRNA, we see a 16% reduction in runtime, but here we have a slight decrease in sensitivity, from 0.87 to 0.86.

5. CONCLUSIONS AND FUTURE WORK

Our novel method improves the overall performance of Dynalign for both the tRNA and 5sRNA families simply by changing the alignment search space restriction. Runtime was reduced 23% for tRNAs, and 16% for 5sRNAs. Although we demonstrated this technique in Dynalign, the methods employed here could also be integrated into other algorithms for joint prediction of secondary structure across multiple homologs to improve their efficiency.

Future constraints on the alignment search space could separate the concept of percent sequence similarity currently two metrics measuring insertions and mismatches, and use these to adapt parameters. In addition, our current method does not take into account the complete distribution of the estimated probabilities. A better method may be to look at the total probability over a particular region of alignment events, and restrict the search space to ensure that this total probability is suitably low.

6. REFERENCES

[1] Mathews DH, Turner DH: **Dynalign: An Algorithm for Finding the Secondary Structure Common to two sequences**. *J Mol Biol* 2002, **317**:191-203.

[2] Harmanci AO, Sharma G, Mathews DH: **Efficient pairwise RNA structure prediction using probabilistic alignment constraints in Dynalign.** *BMC Bioinformatics* 2007, 8:130.

[3] Eddy SR, Durbin R: **RNA sequence analysis using covariance models**. *Nucleic Acids Research* 1994, **22(11):**2079-2088.

[4] Zuker M, Stiegler P: **Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information**. *Nucleic Acids Research* 1981, **9**:133-148.

[5] Gutell RR, Lee JC, Cannone JJ: **The accuracy of Ribosomal RNA Comparative Structure Models**. *Curr Opin Struct Biol* 2002, 12:302-310

[6] Sankoff D: **Simultaneous Solution of RNA folding, Alignment, and Protosequence Problems**. *SIAM J App Math* 1985, **45(5):**810-825.

[7] Gorodkin J, Heyer L, Stormo G: **Finding the most significant common sequence and structure motifs in a set of RNA sequences**. Nucleic Acids Research 1997, 25(18):3724-3732.

[8] Dowell RD and Eddy SR: **Efficient pairwise RNA structure prediction and alignment using sequence alignment constraints**. *BMC Bioinformatics* 2006, **7**:400.

[9] Holmes I, Rubin GM: **Pairwise RNA structure comparison with stochastic context-free grammars**. *Pac Symp Biocomput* 2002, 163-174.

[10] T. Babak, B.J. Blencowe, T.R. Hughes, "Considerations in the identification of functional RNA structural elements in genomic alignments**,"** *BMC Bioinformatics 2007*, **8**:33.

[11] Griffiths-Jones S, Bateman A, Marshall M, Khanna A, Eddy SR: **RFAM: An RNA Family Database.** Nucleic Acids Res 1998, **26:**148-153

[12] A. D'Orazio, "**Adaptive Determination of Alignment Constraints for RNA Secondary Structure Prediction.**" M.S. Thesis, ECE Dept, University of Rochester, Rochester, NY, 2008.