Finding non-coding RNAs through genome-scale clustering

Huei-Hun Tseng[1], Zasha Weinberg[2], Jeremy Gore[2], Ronald R. Breaker[2,3,4] and Walter L. Ruzzo[1,5]

Departments of [1] Computer Science and Engineering and [5] Genome Sciences
University of Washington,
Seattle, WA 98105, USA
E-mail: {lachesis,ruzzo}@cs.washington.edu

[2] Department of Molecular, Cellular and Developmental Biology, [3] Howard Hughes Medical Institute, [4] Department of Molecular Biophysics and Biochemistry
Yale University,
New Haven, CT 06520-8103, USA
E-mail: zasha.weinberg@yale.edu

Non-coding RNAs (ncRNAs) are transcripts that do not code for proteins. Recent findings have shown that RNA-mediated regulatory mechanisms influence a substantial portion of typical microbial genomes. We present an efficient method for finding potential ncRNAs in bacteria by clustering genomic sequences based on homology inferred from both primary sequence and secondary structure. We evaluate our approach using a set of Firmicutes sequences, and the results show promise for discovering new ncRNAs.

Keywords: noncoding RNA, RNA discovery, hierarchical clustering, motif discovery

1. Introduction

1.1. *Motivation and Related Work*

Non-coding RNAs (ncRNAs) are functional transcripts that do not code for proteins. Recent findings have shown that RNA-mediated regulatory mechanisms influence a substantial portion of typical microbial genomes,[1] drawing increasing attention to their study. A major approach for computational detection of ncRNAs is through comparative genomics,[2] where conserved structures are predicted from sequences of multiple species. The key difficulty with this approach is that homologous ncRNAs are often divergent because compensatory mutation preserves structure while changing the sequence. Unfortunately, existing ncRNA-discovery algorithms that consider secondary structure are impractical for genome-scale searches since they are computationally expensive, and work best when applied to datasets in which homologous ncRNAs predominate. Together, these considerations suggest the following strategy: gather clusters of sequences so that each cluster is sufficiently small and enriched in homologous elements for successful computational motif prediction.

Recently, Yao *et al.*[3] applied this strategy to search for bacterial *cis*-regulatory RNAs. Because *cis*-regulatory RNAs are often upstream of genes, they clustered regions upstream of homologous genes (a "gene-oriented" approach). They avoided the need for accurate alignment by using a tool called CMFinder[4] that can predict RNA motifs in unaligned sequences in the face of low sequence conservation, extraneous flanking regions and unrelated sequences. The method successfully recovered most known Rfam[5] families in Firmicutes. Coupled with careful manual evaluation of top-ranking results, this paper and Weinberg *et al.*[6] identified 29 novel RNAs including several riboswitches,[7,8] some of which have been experimentally validated. However, this approach will detect ncRNAs only if they are well-represented upstream of homologous genes. For example, ncRNA genes that are independently transcribed (e.g., SRP, RNaseP, tRNAs) will tend to maintain particular neighboring genes only through a narrow phylogenetic range. This is true of some ncRNAs in the Firmicutes (and Yao *et al.* generally recovered these), but others will be missed. Another important example of the ncRNAs that might be missed by a gene-oriented approach are ones that regulate several non-homologous genes in a phylogenetically narrow range of species.

The main contribution of this paper is the development of an "IGR-oriented pipeline" that clusters intergenic regions (IGRs) based on a combination of sequence and structure similarity, independent of gene context, for purposes of ncRNA discovery. We believe it can identify ncRNAs that are difficult to find with a gene-oriented strategy. For example, an early version of our IGR-oriented approach (unpublished data) correctly predicted 7 related riboswitches regulating purine biosynthesis genes in *Mesoplasma florum*[9] with no close relatives in other sequenced species, exactly the second scenario outlined above.

1.2. *Efficient pipeline for detecting ncRNAs*

To be able to detect ncRNAs computationally, we wish to identify homologous RNA sequences. To do this without gene context, we search through entire intergenic regions (IGRs) of several species for homology. Homologous ncRNAs usually exhibit some conservation in primary sequence, but detection of this similarity is often impossible without exploiting the significant conservation of RNA structure. Traditional structure-based methods[10] perform well but are extremely slow, making them impractical for large search spaces. We design a novel lightweight approach that incorporates both secondary structure information and primary sequence homology via BLAST (referred to as the *folded*-BLAST approach). The goal is to achieve the best sensitivity possible, while maintaining feasible search time.

We wish to group sequences based on homology relationships. However, RNAs may contain multiple domains with sequence homology recognizable by BLAST, but these domains may be separated by dissimilar regions. To account for this, we design a hierarchical clustering method that, given a set of pairwise homology hits, heuristically merges and clusters overlapping sequences. Finally, as in Yao *et al.*'s

pipeline,[3] the clusters can be used to predict motifs, which in turn can be used to scan genomes for more motif instances (motif scan).

Our proposed pipeline for a given input set of genomic sequences, then, consists of the following steps: (1) intergenic region extraction; (2) homology search; (3) hierarchical clustering; (4) motif discovery; and (5) motif scan.

Our pipeline shares high level goals with the work of Will *et al.*,[11] but differs in emphasis, and is somewhat complementary to it. Both cluster intergenic sequences based on homology, then attempt to predict RNA motifs in these clusters. Will *et al.*, building on Missal *et al.*,[12] need reliable sequence alignments for their motif prediction step, so they use a stringent BLAST E-value threshold for this phase. To recover broader RNA families, they apply a second clustering step to cluster the RNA motifs produced in the first step. The number of RNA predictions is much smaller than the number of IGRs, and they can afford to apply sophisticated but computationally expensive structure-based clustering methods here, and their paper develops such a method (LocARNA). In contrast, we use an RNA motif prediction tool that tolerates unaligned inputs and thus can be more aggressive in trying to gather more (and more remote) homologs, on the premise that more examples will allow inference of more accurate models. Hence, we cluster intergenic sequences based on relatively permissive BLAST searches. A novelty of our approach is incorporating secondary structure information in this clustering stage. Neither method attempts direct pairwise structure comparison among all intergenic sequences; that appears prohibitively expensive on data sets of this scale.

1.3. *Evaluation*

We clustered a set of Firmicutes genomic sequences, and evaluated them using a set of ncRNA families mainly consisting of riboswitches. Riboswitches are metabolite-sensing RNAs that regulate gene activity through binding to ligands and modifying the expression of biosynthetic and transport proteins for those ligands.[7,8] They are structurally conserved with an average family sequence identity of 55–80% and average sequence length of 60–200 nts. Primary sequence-only methods captured ∼84% of the known ncRNAs, with an average cluster specificity of ∼40%. Incorporating secondary structure captures about 80% of the known ncRNAs while increasing average cluster specificity to ∼50%. Motifs predicted from the ncRNA-containing clusters were then used to scan a test set, and the *folded*-BLAST approach achieved median sensitivity of 76% with 99% specificity, much better than the best primary sequence-only approach (sensitivity 61%). Moreover, several motifs predicted from *folded*-BLAST clusters were more similar, and in some cases, almost identical to trusted riboswitch models. This suggests that our novel method of secondary structure-incorporation enhances clustering, which in turn increases the likelihood of inferring a strong motif.

2. Results

Full genomic sequences from a set of 212 Firmicutes species were used as input. The entire set contains 1252 known ncRNAs, 1008 of which are completely covered by our extracted intergenic regions. Primary sequence homology data were obtained using NCBI-BLAST,[13] WU-BLAST,[14] or SSEARCH.[15] To incorporate structure into our homology searches, we used WU-BLAST, since it allows convenient usage of arbitrary scoring matrices.[a]

2.1. *Clustering evaluation*

Table 1 shows evaluation for the Firmicutes clusters generated by our pipeline. Note that per-cluster specificity (p) is only a lower-bound, since unannotated members of a cluster could be undiscovered ncRNAs. NCBI-BLAST generally has the best capture count, i.e., clustering the largest number of known ncRNAs in any cluster, yet also the worst per-cluster specificity. *folded*-BLAST captures fewer, yet average cluster specificity generally tops all the rest. However, no program in Table 1 consistently surpasses the others.

Since our goal is to detect novel ncRNA families, we turn our attention to individual clusters with good specificity. For example, *folded*-BLAST produced 16 clusters that contained at least one TPP riboswitch, one of which had specificity of 35/39, another 10/10, and another 7/9. If any of these could yield a representative motif, then a motif scan would likely recover other TPP riboswitches. Thus, what is more important is our ability to produce clusters that permit RNA alignment tools like CMFinder to correctly predict structured RNAs. In the following section, we show results of predicted motifs from selected clusters.

2.2. *Motif discovery and scanning*

CMFinder predicts zero or more motifs in all ncRNA-containing clusters. For any motif, along with the covariance model (CM) produced, we do a CM scan if the number of cluster members containing this motif is at least 6, and that the average motif score (generated by CMFinder) is at least 50. These criteria are set because weak motifs/CMs will likely introduce false hits. The CMs scan our entire ncRNA dataset: $\sim$1 Mb of ncRNAs from all available bacterial species (not just Firmicutes), plus a control set of $\sim$5 Mb of randomly selected IGRs (from various species) not containing known ncRNAs.

In this CM scan test, a hit is considered correct only if it matches a selected ncRNA on the correct strand. Strictly speaking, we cannot be sure that our randomly selected IGRs indeed do not contain any undiscovered ncRNAs, but for the purpose of evaluation, we assume there to be none.

[a]Details are in Methods. Supplementary materials including additional method details and results are available at: http://bio.cs.washington.edu/supplements/lachesis/APBC2008.

Table 2 summarizes the individual CM scans recovering the most instances for each particular family. For most of the more abundant families, such as FMN, SAMI, TPP, ykoK and ydaO riboswitches, all four programs had a best CM scan of ~ 0.9 sensitivity with ~ 1.0 specificity. Recovery of purine riboswitches was consistently low, and we observed that it was because non-Firmicutes purine riboswitches have much longer single-stranded terminal regions than their Firmicutes counterparts. Of particular interest are the 7 *Mesoplasma florum* purine riboswitches, a difficult case for gene-oriented pipelines. In NCBI-BLAST, 6 of the 7 were grouped in a cluster of size 44 (6/44), and although CMFinder succeeded in producing a representative motif, it had low specificity: the CM scan reported back the 6 on which it was trained, along with almost 2000 false hits. The motif discovered from the WU-BLAST cluster was more specific, but still had 96 false hits. In contrast, SSEARCH generated a 6/7 cluster and the resulting CMfinder motif scans reported back the 6 with no false hits. *folded*-BLAST produced a 7/32 cluster, but CMFinder did not predict a motif, so no CM scan was done. We examined why CMFinder failed in

Table 1. Clustering evaluation by individual ncRNA families

ncRNA (#)	RfamID or ref.	avg. seq. len. (nts)	*c*: captured count *p*: avg. cluster specificity							
			NCBI-BLAST		WU-BLAST		SSEARCH		*folded*-BLAST	
			c	*p*	*c*	*p*	*c*	*p*	*c*	*p*
t-box (452)	RF00230	223	**441**	0.42	406	0.49	423	0.51	352	**0.60**
SAMI (113)	RF00162	124	**104**	0.44	99	0.57	100	**0.59**	102	0.55
TPP (90)	RF00059	97	**77**	0.29	70	0.21	72	0.39	75	**0.61**
purine (66)	RF00167	99	**64**	0.11	52	0.23	52	**0.30**	59	0.21
ylbH (53)	RF00516	143	**37**	0.08	27	0.11	28	**0.13**	26	0.06
cobalamin (51)	RF00174	201	**51**	0.51	46	0.61	45	0.22	45	**0.67**
lysine (46)	RF00168	181	**36**	0.17	33	**0.45**	34	0.41	33	**0.45**
SRP (41)	RF00169	99	**41**	0.42	40	0.37	39	0.41	**41**	**0.45**
RNaseP (40)	RF00011	360	**37**	0.77	**37**	0.82	**37**	0.39	**37**	**0.86**
FMN (40)	RF00050	147	**40**	0.59	**40**	**0.82**	**40**	0.75	**40**	0.67
glycine (38)	RF00504	90	32	0.29	30	**0.58**	**33**	0.41	28	0.29
preQ1 (37)	RF00522	69	**33**	0.15	18	0.24	23	**0.42**	19	0.39
ydaO (29)	RF00379	171	**29**	0.85	27	**0.95**	26	0.81	24	0.86
yybP (29)	RF00080	131	**28**	0.34	23	0.36	26	0.34	23	**0.44**
6S (26)	RF00013	205	23	0.32	**24**	0.32	22	0.56	**24**	**0.66**
ykoK (25)	RF00380	178	**25**	0.29	**25**	0.46	22	0.51	**25**	**0.96**
glmS (19)	RF00234	183	18	0.46	18	**0.86**	18	0.29	**19**	0.34
ykkC (12)	RF00442	129	**10**	0.46	**10**	0.47	8	0.35	9	**0.50**
moco (10)	Weinberg et al.[6]	132	6	**0.52**	6	0.51	3	0.21	**7**	**0.52**
SMK (10)	Fuchs et al.[16]	173	**6**	0.38	2	0.30	**6**	**0.83**	2	0.26
Median**			**0.89**	0.34	0.79	0.46	0.85	0.41	0.79	**0.51**

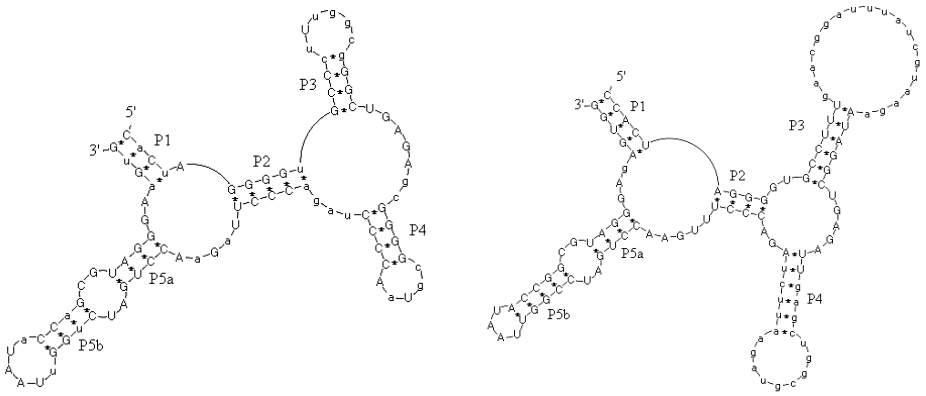
Note:
For each ncRNA family F we give its name, the number of members covered by the extracted IGRs of our Firmicutes test set, the Rfam accession or other reference, the average length of members, and performance statistics for the clustering methods. For each F, we report c, the *captured count*, i.e., the number of ncRNAs in F covered by a member in some cluster, and p, the *cluster specificity*, i.e., the average over all clusters C containing members of F, of the percentage of members of F in C. We define a known ncRNA as "covered" by a IGR segment if the segment covers at least 50 nts or 50% of the ncRNA region.
**: For c, the median is taken over the ratios of capture count to family size.

Table 2. CM scan best recovery motif comparison. For each ncRNA family and each homology search program used, the motif/CM that recovered the most instances of the particular family is listed. The actual motif identifications can be cross-referenced online. *sen.* is the recovery percentage (sensitivity), and *spe.* is the specificity of the CM scan; "None" indicates that no instances were recovered.

ncRNA	NCBI-BLAST		WU-BLAST		SSEARCH		*folded*-BLAST	
family	*sen.*	*spe.*	*sen.*	*spe.*	*sen.*	*spe.*	*sen.*	*spe.*
t-box	0.69	0.98	**0.71**	**0.99**	0.41	**0.99**	0.68	**0.99**
SAMI	0.94	**0.99**	**0.96**	**0.99**	0.84	**0.99**	0.94	**0.99**
TPP	0.84	**0.99**	0.95	**0.99**	0.54	**0.99**	**0.96**	**0.99**
purine	0.36	**0.99**	0.36	**0.99**	0.32	**0.99**	**0.37**	**0.99**
ylbH	0.01	0.5	0.01	**1.00**	**0.02**	**1.00**	0.01	0.33
cobalamin	None		0.84	0.82	0.72	**1.00**	**0.86**	0.99
lysine	0.79	**1.00**	**0.84**	0.82	0.72	**1.00**	0.74	**1.00**
SRP	0.1	0.99	0.1	**1.0**	**0.84**	0.98	0.77	0.98
RNaseP	**1.0**	**0.99**	**1.0**	**0.99**	**1.0**	**0.99**	**1.0**	**0.99**
FMN	**0.96**	**0.99**	**0.96**	**0.99**	**0.96**	**0.99**	**0.96**	0.98
glycine	None		0.08	0.98	None		**0.86**	**0.99**
preQ1	**0.01**	**0.04**	None		**0.01**	0.02	None	
ydaO	**0.97**	**1.00**	0.96	**1.00**	0.96	**1.00**	0.96	0.99
yybP	**0.26**	**1.00**	0.11	**1.00**	0.11	**1.00**	0.22	0.99
6S	0.09	**1.00**	**0.42**	0.92	0.09	**1.00**	0.29	**1.00**
ykoK	**0.96**	**1.00**	**0.96**	**1.00**	0.90	**1.00**	**0.96**	**1.00**
glmS	**0.95**	**1.00**	0.93	**1.00**	0.91	**1.00**	**0.95**	**1.00**
ykkC	None		None		**0.69**	**1.00**	**0.69**	**1.00**
moco	None		None		None		None	
SMK	**0.08**	**0.67**	None		None		None	
Median	0.31	**0.99**	0.56	**0.99**	0.61	**0.99**	**0.76**	**0.99**

folded-BLAST's case, and determined that CMFinder's prior parameter for the expected fraction of motif-containing instances was higher than the actual percentage. If the percentage was lowered from 0.5 to 0.2, CMFinder would find a representative motif for 6 of the *M. florum* purine riboswitches. However, lowering the percentage might entail tradeoffs.

Fig.1(a) depicts the motif recovering the most TPP riboswitches using the *folded*-BLAST approach, while Fig.1(b) is the (hand curated) consensus motif from the Rfam TPP seed alignment. The *folded*-BLAST motif has a longer unpaired region on the $5'$ end, but shares an almost identical structure and base composition with Rfam's. It is encouraging to see how similar the two structures are, given that the cluster used to predict motif Fig.1(a) had 30 sequences, and Fig.1(b) was constructed out of 174 seed sequences. The best TPP-recovering motif produced by the other three programs (not drawn due to space limit) correctly predicted most of the structure, but varied in $5'$ and $3'$ ends: WU-BLAST's does not have the closing stem loop, resulting in a much longer unpaired region on both ends; NCBI-BLAST's missed both the closing stem loop and part of a multiloop; SSEARCH's predicted most of the base pairings correctly, but had a 70-nts $5'$ unpaired region, which is probably why the CM scan recovery was poor. We noticed that the TPP-containing sequences in the best-TPP-recovery clusters for NCBI-BLAST and SSEARCH were



(a) *folded*-BLAST (b) Rfam

Fig. 1. A TPP riboswitch motif automatically predicted by our pipeline *vs* the (hand-curated) Rfam structure. Terminal single-stranded regions are not shown for simplicity. (a) Motif predicted using *folded*-BLAST approach that resulted in the best CM scan recovery of TPP riboswitches. (b) Consensus motif from the Rfam TPP riboswitch seed alignment (Rfam id RF00059).

50-200 bps longer than the ones in WU-BLAST and *folded*-BLAST, and it is possible that we had tuned the parameters for the first two programs in such a way that IGR fragments are easily joined together into long sequences during clustering pre-processing.

Two small ncRNA families, the moco and preQ1 riboswitches, had poor CM scan recoveries. The moco riboswitch was discovered by manual inspection of CMfinder motifs from the gene-oriented pipeline,[6] but is not common in Firmicutes, making motif discovery difficult, even though all four homology search programs produced good clusters, grouping 5 or 6 instances in compact clusters of size 5 or 6. For preQ1, though it has more Firmicutes instances, is short (65 nts on average), which made accurate and compact clustering challenging.

3. Methods

3.1. *Extracting intergenic regions (IGR)*

Given an input genomic sequence, we remove regions annotated in RefSeq as coding regions, repeat regions, tRNAs or rRNAs. Both strands are removed when one strand contains one of the above annotations. This breaks a genomic sequence into a set of intergenic regions (IGRs). We then discard all IGRs shorter than 15 nts along with those immediately adjacent to an annotated rRNA region, for we find their $5'$ and $3'$ borders to be frequently misannotated.

Removing genomic regions encoding for genes or known RNA elements on either strand reduces search space, yet might risk missing ncRNAs. Using our ncRNA dataset, we examine how much will be missed in our Firmicutes set, and how much can be gained by extending our search space into annotated regions. Our dataset contains 1236 Firmicutes ncRNAs, and if we hypothesize that a region containing

a ncRNA will have a chance of being grouped with other homologous regions if and only if there exists an extracted IGR that covers at least 50% or 50 nts of the ncRNA, then we will miss 107 ncRNAs. Even if we extend our extracted IGRs 200 nts on both ends, almost doubling the search space, by our hypothesized definition we still miss 59. We have several explanations for this: If a ncRNA overlaps an annotated coding region, the RefSeq record could have mis-annotated the location. Also, ncRNAs might overlap other functional regions either due to the evolutionary pressure of keeping genomes compact, or because their mechanism of gene regulation requires some overlap. For simplicity in this study, we do not extend IGRs.

3.2. *Homology search*

To compare performance, we used several popular search programs, including NCBI-BLAST, WU-BLAST, and SSEARCH. SSEARCH[15] implements the Smith-Waterman local alignment algorithm; it is 10 times slower than BLAST programs, but is thought to be more sensitive. NCBI-BLAST and WU-BLAST are both heuristic approximations to Smith-Waterman, and begin alignment by matching exact short words (*seeds*). In this study, we use a seed length of 11 because preliminary tests indicated that it has reasonable sensitivity and speed.

3.3. *Homology search with predicted secondary structure*

To implement *folded*-BLAST, we use RNALfold from the Vienna package[17] to compute locally stable RNA secondary structures with a maximal base span L (empirically set to 200). Given an input sequence and a defined L, RNALfold lists predicted secondary structure components. However, since it has been shown that secondary structure alone is insufficient for detecting ncRNAs,[18] we cannot entirely trust the boundaries and structures predicted. Hence, we developed a heuristic procedure to merge RNALfold's components, breaking long IGRs into small, overlapping pieces with lengths of 200-500 nt. For each piece, RNAfold predicts whether each nucleotide is paired upstream, paired downstream or unpaired. To take advantage of fast primary sequence homology search programs, we map these sequences into a 12 letter alphabet representing nucleotide plus pairing direction. The resulting sequences are treated like protein sequences, but we search using a handmade scoring matrix in which nucleotide identity (match) is favored, but when the predicted structures are the same, nucleotide mismatch penalty is mitigated. The matrix is detailed in the online supplement.

3.4. *Clustering*

Prior to clustering, we pre-process pairwise homology hits obtained from homology search programs into *nodes* and *edges*. A node represents an IGR segment, and an edge represents a homology hit. For all homology search results, hits with score less than 40 or greater than 300 are ignored. Note that, although the same score may

have different statistical significance depending on the program used, we have tuned the parameters to achieve statistical similarity, and have observed that all programs generally produce the same distribution of scores. For *folded*-BLAST, an additional criterion is added: hits with percentage identity less than 0.3 or positive percentage identity (percentage of alignments contributing positive scores) less than 0.5 are ignored. The cutoff values were determined using small test sets of ncRNAs against random sequences.

When we process a homology hit, we first check whether there already is a node representing a segment overlapping the query region by 15 nts. If so, then that node is expanded to represent the union of the two regions; otherwise a new node is created. The same procedure is applied to the aligned subject region. We then create an edge to represent the hit, whose weight is the homology score. In sum, the output of pre-processing a set of homology hits is a weighted, undirected graph.

The clustering step uses WPGMA (Weighted Pair Group Method using Arithmetic averaging), also known as average-linkage clustering. Edge weights are used as scores. Missing edges are assumed to have score 0.

The output of the hierarchical clustering is a forest of trees. Some trees can be as small as only 2 leaves, which means that the homology search program did not find any other IGR segments significantly homologous to them. The largest tree can be as large as the number of nodes. Such a supersized cluster is impractical for any further evaluation, and given that most of our ncRNA families have no more than 100 instances in our species sets, we generally use a size cutoff of 50. More adaptive tree-cutting is discussed below.

3.5. *Motif prediction and scan*

Motif prediction and scan are done as in Yao *et al.*, 2007, excluding the (subjective) manual evaluation steps. Briefly, CMfinder[4] folds each sequence in its input set, and constructs an initial heuristic alignment attempting to match similar sequence and structural features between sequences. Next it builds a covariance model (CM) from the alignment, exploiting both mutual information and single-sequence structure predictions to arrive at a consensus structure prediction. Finally, it performs an EM-like iteration, alternately realigning the sequences to the model and rebuilding the model from the refined alignment. It is robust to non-motif containing sequences and extraneous regions flanking the motifs. Parts of CMfinder use the Infernal[19] software package, which was also used for the scanning step in our evaluation. On larger data sets, we would also use the RaveNnA[20] filtering package.

4. Discussion and Future Work

Refining the design of our ncRNA discovery pipeline is complicated because there is no clear winner among applicable homology and motif tools. We plan to improve our pipeline in various aspects, particularly the following: (1) Secondary structure incorporation: The heuristics used in our novel method for incorporating secondary

structure were empirically determined. For example, secondary structures were predicted with a maximal base span of 200, and the scoring matrix used for *folded*-BLAST was handmade (we found scores trained from curated ncRNA alignments to perform poorly). (2) Hierarchical clustering: We merged overlapping homologous sequences based on the assumption that evolutionary divergence causes homology search programs to fail to capture full length homologous ncRNAs. We have neither deeply investigated this assumption nor determined an optimal merging strategy. (3) Adaptive tree cutting: Our fixed size-cut for partitioning large clusters may compromise motif prediction for some ncRNA families. For example, *folded*-BLAST clustered the 7 *M. florum* purine riboswitches into a compact subbranch, yet the 50 size-cut included extraneous sequences that caused CMFinder to fail in predicting a motif. To improve specificity, we could try using other evidence (e.g., homology scores) to trim a cluster, or iteratively use CMFinder to add or remove members until all cluster members are predicted as containing motif instances.

With future evaluation on other species and improvement of the existing pipeline, we hope to identify and experimentally verify novel structured RNAs.

References

1. W. C. Winkler, *Curr Opin Chem Biol* **9**, 594(Dec 2005).
2. E. Rivas and S. R. Eddy, *BMC Bioinformatics* **2**, p. 8 (2001).
3. Z. Yao, J. Barrick, Z. Weinberg, S. Neph, R. Breaker, M. Tompa and W. Ruzzo, *PLoS Comput Biol* **3**, p. e126(July 2007).
4. Z. Yao, Z. Weinberg and W. L. Ruzzo, *Bioinformatics* **22**, 445(February 2006).
5. S. Griffiths-Jones, A. Bateman, M. Marshall, A. Khanna and S. R. Eddy, *Nucleic Acids Res* **31**, 439 (2003).
6. Z. Weinberg, J. Barrick, Z. Yao, A. Roth, J. Kim, J. Gore, J. Wang, E. Lee, K. Block, N. Sudarsan, S. Neph, M. Tompa, W. Ruzzo and R. Breaker, *Nucleic Acids Res* **35**, 4809(July 2007).
7. R. L. Coppins, K. B. Hall and E. A. Groisman, *Curr Opin Microbiol* **10**, 176(Apr 2007).
8. B. J. Tucker and R. R. Breaker, *Curr Opin Struct Biol* **15**, 342(Jun 2005).
9. J. Kim, A. Roth and R. Breaker, *Proc Natl Acad Sci U S A* **104**(Oct 2 2007).
10. E. K. Freyhult, J. P. Bollback and P. P. Gardner, *Genome Res* **17**, 117(Jan 2007).
11. S. Will, K. Reiche, I. L. Hofacker, P. F. Stadler and R. Backofen, *PLoS Comput Biol* **3**, p. e65(Apr 2007).
12. K. Missal, X. Zhu, D. Rose, W. Deng, G. Skogerbo, R. Chen and P. F. Stadler, *J Exp Zoolog B Mol Dev Evol* **306**, 379(Jul 2006).
13. S. Altschul, W. Gish, W. Miller, E. Myers and D. Lipman, *J Mol Biol* **215**, 403 (1990).
14. W. Gish, (1996-2004), `http://blast.wustl.edu`.
15. W. R. Pearson, *Methods in Molecular Biology* **132**, 185 (2000).
16. R. T. Fuchs, F. J. Grundy and T. M. Henkin, *Nat Struct Mol Biol* **13**, 226(Mar 2006).
17. I. L. Hofacker, W. Fontana, P. F. Stadler, L. S. Bonhoeffer, M. Tacker and P. Schuster, *Monatshefte für Chemie* **125**, 167 (1994).
18. E. Rivas and S. R. Eddy, *Bioinformatics* **16**, 583 (2000).
19. S. R. Eddy, *BMC Bioinformatics* **3**, p. 18 (2002).
20. Z. Weinberg and W. L. Ruzzo, *Bioinformatics* **22**, 35(January 2006).