Annotating functional RNAs in genomes using Infernal

Eric P. Nawrocki

Abstract Many different types of functional non-coding RNAs participate in a wide range of important cellular functions but the large majority of these RNAs are not routinely annotated in published genomes. Several programs have been developed for identifying RNAs, including specific tools tailored to a particular RNA family as well as more general ones designed to work for any family. Many of these tools utilize covariance models (CMs), statistical models of the conserved sequence and structure of an RNA family. In this chapter, as an illustrative example, the Infernal software package and CMs from the Rfam database are used to identify RNAs in the genome of the archaeon *Methanobrevibacter ruminantium*, uncovering some additional RNAs not present in the genome's initial annotation. Analysis of the results and comparison with family-specific methods demonstrate some important strengths and weaknesses of this general approach.

Key words: covariance models; Infernal; Rfam; stochastic context-free grammars; homology search; genome annotation; tRNA; rRNA; SRP RNA; RNase P RNA; CRISPR; riboswitch;

1 Introduction

Genome annotation is the identification of functional sequence elements in an organism's genome. Knowledge of the presence and location of these sequence elements coupled with understanding of their functional roles helps reveal the types of biological processes that take place in the organism as well as the evolutionary history of that organism. Classes of functional sequence elements include protein-coding genes, non-coding RNA elements, promoter elements, enhancers, as well as others.

Eric P. Nawrocki
Janelia Farm Research Campus, Howard Hughes Medical Institute, 19700 Helix Drive, Ashburn VA 20147, USA, e-mail: nawrockie@janelia.hhmi.org

Functional RNA elements are RNAs that are not translated into proteins, but rather carry out their biological function directly as RNAs. Much like proteins, many of these RNAs fold into a specific three-dimensional structure that is integral to their function. For convenience, I will refer to functional RNAs as simply RNAs in this chapter.

RNAs play vital roles in many cellular processes. For example, transfer RNAs (tRNAs) and ribosomal RNAs (rRNAs) play central roles in the translation of messenger RNAs into proteins. Spliceosomal RNAs (such as U1, U2, U4, U5 and U6) interact with proteins as part of a ribonucleoprotein complex (RNP) responsible for splicing introns from many eukaryotic pre-mRNAs [11]. Small nucleolar RNAs (snoRNAs) are members of RNPs that guide post-transcriptional modification during the maturation of rRNAs and other RNA genes in archaea and eukarya [24]. The SRP (signal recognition particle) RNA is part of an RNP involved in transporting proteins within cells [48]. Ribonuclease P (RNase P) RNA is a vital part of an RNP that processes precursor tRNAs through cleavage of a 5' leader sequence [26].

Many other RNA elements play key roles in gene regulation, such as microRNAs (miRNAs) that act by binding to specific target mRNAs in eukaryotes via basepairing, affecting the expression of the target [12]. Riboswitches are structured RNA elements typically occurring in the 5' untranslated region (UTR) of protein-coding genes over which they exert translational or transcriptional control through binding of small metabolites, which cause a structural change in the riboswitch. They often control genes involved in the transport or biosynthesis of their target metabolite [35]. The bacterial 6S RNA promotes more general gene regulation by binding directly to RNA polymerase and repressing its activity during stationary phase of bacterial growth [71].

Other RNA elements are important for defending cells against viruses and transposons. Small-interfering RNAs (siRNAs) are 21-25 nucleotide long RNAs in eukaryotes often derived from exogenous RNAs that are recognized by the protein complex RISC, ultimately leading to degradation of the exogenous RNA [55]. In archaea and bacteria, a similar defense system is encoded in CRISPR (Clustered Regularly Interspaced Short Palindromic Repeats) elements, which are short 24-48 nucleotide repeats which have been predicted to form hairpin structures separated by similar length spacers of foreign DNA from past exposures to parasites such as viruses (phage). RNAs from CRISPR elements are constitutively expressed and guide silencing of complementary foreign DNA or RNA [38].

The phylogenetic distribution of different types of RNAs varies. Some, such as those involved in ubiquitous cellular processes like tRNAs, rRNAs, RNAse P RNA and SRP RNA, exist in all three domains of life. Others are widespread in one or two of the domains but not the other(s), such as snoRNAs in archaea and eukarya, 6S RNA in bacteria, and microRNAs in eukarya. Finally, some exist within only a specific clade of a domain, ranging in size anywhere from a phylum (e.g. SmY RNAs, found in RNPs predicted to be involved in trans-splicing in nematodes [40]) down to only a few species (e.g. OxyS, a global regulator induced in response to oxidative stress in certain members of one family of gammproteobacteria, including *E. coli* [3]).

Despite the widespread importance of functional RNAs, the large majority of them are typically not annotated in published genomes, whereas it is generally accepted that most protein-coding genes are. This discrepancy is at least partially due to some unique challenges in identifying RNAs. In this chapter, I will discuss this issue in the context of genome annotation using freely available software programs. I will first compare and contrast protein and RNA annotation in genomes, highlighting challenges specific to RNAs, and then some current methods for RNA annotation will be introduced. I will then focus on one specific method: the use of covariance models (CMs) and detail the strengths and weaknesses of the CM approach through a practical example of RNA annotation of an archaeal genome. Finally, I will compare and contrast the general CM-based approach with the use of family-specific tools designed to identify homologs of particular RNA family.

1.1 Genome annotation of proteins and RNAs

The annotation of protein-coding genes in a genome typically consists of two major steps:

Step 1. Predict protein-coding gene sequences.
Step 2. Assign putative functional annotation to the predicted proteins using homology-based search tools.

In the first step, only subsequences of the genomes that correspond to open-reading frames (ORFs) need to be evaluated as possible protein-coding genes. This greatly reduces the possible search space relative to all possible subsequences of the genome, and represents an important distinction between protein annotation and functional RNA annotations of genomes, for which there is no analog of the open-reading frame signal. For archaea and bacteria, accurate programs exist for performing step 1, such as the popular Glimmer program, which can correctly identify about 99% of protein-coding genes with known functions [14]. Similar programs have been designed for eukaryotic genomes including Genscan [10], GeneID [33], and Genemark [50], among others, but these are generally less accurate due to the higher complexity of eukaryotic genomes versus archaeal and bacterial genomes.

Given a set of predicted protein-coding genes, step two aims to functionally annotate these genes based on previous functional annotation of genes with similar sequences, which are predicted to be homologous. This step is carried out using homology search tools like HMMER [20] or BLASTP [2] to search various target databases such as Pfam [25], COG [67], the NCBI NR database [65], and others.

As mentioned, unlike proteins, functional RNAs are not contained within open reading frames and so a different strategy is required for their identification. Many RNAs do, of course, contain signals in their promoter regions that can, and have been, exploited when searching for RNAs [4], but these are often clade- or even species-specific, hampering any general approach by requiring specific foreknowledge of the genome being studied. However, there are several RNA genefinder pro-

grams that attempt to address the RNA analog of step 1 of the protein-coding gene scheme by identifying regions that conserve a statistically significant secondary structure, indicating a structural RNA gene. However, these programs are much less reliable than protein genefinders and in particular suffer from high false positive rates [5, 56, 32] limiting their utility for RNA annotation.

Consequently, RNA annotation is typically performed using known RNAs as queries for homology searches against the entire genome being studied. This is similar to step 2 of protein annotation from above, but homology searches for RNAs are less powerful than for proteins for several reasons: RNAs tend to be shorter than proteins (often about 100 nucleotides, as opposed to 200-300 amino acids [8]), and the search must be carried out at the RNA/DNA level instead of at the protein level, which reduces statistical power due to the smaller alphabet size and the degeneracy of the genetic code [61]. To cope with the reduced statistical signal, the most successful RNA homology search programs take advantage of the structural conservation of many functional RNAs by scoring a combination of the conserved sequence and the secondary structure of an RNA family [27]. Many basepaired nucleotides in a conserved RNA structure tend to covary over evolutionary timescales to maintain complementarity, often by changing from one Watson-Crick basepair to another (A:U or C:G) or to a G:U wobble basepair. This covariation offers a useful statistical signal that can be used in addition to sequence conservation when searching for homologous structural RNAs. Sequence- and structure-based tools can be divided into two classes: family-specific methods that are designed for a particular RNA family, and general tools that can work for any family.

1.1.1 Family-specific RNA search methods

Table 2 summarizes some popular family-specific programs for identifying RNAs in genomes. The most widely used RNA homology search tool targets the single largest gene family, tRNAs. The tRNAscan-SE program [51] uses a powerful statistical model called a covariance model (CM) that scores candidates based on both their sequence and predicted secondary structures. CMs are more sensitive (able to find more true homologs) than sequence-only based searches but are much slower, due to the higher complexity of their scoring algorithms (as discussed in section 1.2) to the point of being impractical when searching large sequence databases. CMs outperform sequence-based methods particularly well for families like tRNA that are short (about 70 nucleotides) and exhibit low levels of sequence similarity while maintaining a highly conserved secondary structure. To deal with the slow search speed of CMs, tRNAscan-SE uses fast tRNA-specific prefilters that remove a large fraction of the database, leaving only promising subsequences to be evaluated by the slow CM methods. The result is a tool fast enough to search large mammalian genomes on a desktop computer in a few hours.

The strategy of using fast family-specific filters prior to a CM based search is employed by other family-specific RNA search tools. For example, the Bcheck program [76] uses the sequence and structure based pattern matching program RNABOB [18]

as a fast prefilter for CMs to identify RNase P genes. RNABOB, like other pattern matching programs, identifies subsequences that can fold into a particular structure based on user-defined constraints. SRPscan [63] uses the same strategy with signal recognition peptide (SRP) RNA patterns and CMs to identify SRP RNAs.

Other sequence- and structure-based tools do not use CMs. Aragorn is a tRNA and tmRNA finder [43] that uses a tRNA-specific search algorithm that searches for part of the highly conserved B-box consensus sequence as an alignment seed and expands a structure-aware alignment around that seed. Aragorn's sensitivity is similar to tRNAscan-SE's but it is about an order of magnitude faster. The Arwen program [44] from the developers of Aragorn detects tRNA sequences in mitochondria in a similar manner.

Structure-based methods are not necessary for all RNAs. For example, the small and large subunit ribosomal RNAs (SSU and LSU rRNA) differ markedly from tRNA both in their size (about 1500 nt and 3000 nt, respectively) and high level of sequence conservation, and sequence-based homology search methods perform well for these RNAs. They are sometimes annotated using the pairwise sequence similarity search tool BLASTN [2] with homologous query sequences from closely related species, or with the RNAmmer tool [42] based on sequence-based profile hidden Markov models (discussed in more detail in section 1.2).

Some family-specific methods cannot directly be used to scan genomes. For example, the SnoReport program [36] uses pattern descriptors as filters for a SVM-based classification, but requires the target sequences be short candidate snoRNAs, not genome-length sequences.

With the exception for tRNAscan-SE and RNAmmer, none of these tools are commonly used for annotating a genome prior to its publication in a database, as demonstrated by a sampling of fourteen published genomes (five archaea, five bacteria and four eukarya) in NCBI's GenBank database shown in Table 1. Notably, for one of the bacterial genomes listed, *Citrobacter rodentium*, 56 RNAs other than tRNAs and rRNAs were annotated using the Infernal software package and the Rfam database. Infernal implements general CM search methods that can be used for any RNA family, and Rfam contains CMs for about 1500 RNA families. In the remainder of the chapter I will discuss CMs, Infernal and Rfam and their potential for large scale annotation of RNAs in genomes.

1.2 Covariance models

Covariance models (CMs) are probabilistic models of the sequence and secondary structure of an RNA family [23, 15]. They are constructed from multiple sequence alignments of known homologs of the family that are annotated with a consensus secondary structure. A CM is useful for searching databases for homologs of the family it models, and for creating sequence- and structure-based multiple sequence alignments of those homologs.

CMs are stochastic context-free grammars (SCFGs), introduced in chapters 5 and 8 of this book. More specifically, they are profile SCFGs, analogous to profile hidden Markov models, commonly used for linear sequence analysis of protein domain families, but with added complexity for modeling a conserved secondary structure. Like CMs, a profile HMM is constructed from an alignment of homologous sequences (but without structure annotation).

The key feature of a profile model is its *position-specificity* [31]: each position of the input alignment is modeled independently. This allows profile methods to take into account the level of conservation at each position when scoring/aligning candidate family members, by defining a scoring system that weighs matches and mismatches at highly conserved positions more than at highly variable positions.

For example, take the toy RNA family depicted in Figure 1, represented by the ungapped alignment of eight RNA homologs of length 11. Imagine a simple sequence-profile model that scans a target database shifting a length 11 window one nucleotide at a time looking for putative homologs of this family. If the target subsequence contains a nucleotide observed in at least one of the eight known homologs at every position then it is considered a match, otherwise it is a mismatch. In this scenario, the specificity of positions is defined by their conservation in the known homologs. For example, alignment positions 4 through 7 are completely conserved, containing UUCG for all eight homologs, while positions 3 and 8 are completely variable (25% A, C, G and U), meaning that a putative homolog must contain UUCG at positions 4 through 7 but can be any nucleotide at positions 3 and 8.

In general, conserved positions are more informative than variable ones as to whether a sequence matches a profile or not. It is possible to quantify the amount of information in a sequence profile based on the alignment the profile was built from. Completely conserved positions contain two *bits* of information[15], because they specify a single choice out of four possible choices, corresponding to answering two yes/no questions to narrow four possibilities down to one. A position that contains two nucleotides, with each at half of the positions, contains one bit of information (e.g. positions 2 and 10 in Figure 1). A completely variable position contains zero bits of information, because any of the four nucleotides will match. The total amount of information in a profile indicates the likelihood of a match to the profile in a random sequence database. For the 14 bit sequence profile corresponding to the alignment in Figure 1, we expect a match once in every $2^{14} = 4096$ nucleotides in a random, so-called *iid* (independent, identically distributed) sequence database in which each nucleotide has an equiprobable chance of being observed at each position (25% chance of being A, C, G, or U).

In the case of structural RNAs, we can increase the information of a profile by considering the conserved consensus structure of the family as well as the conserved sequence, allowing us to better discriminate good matches to the model, which represent putative homologs, from background, nonhomologous sequence when searching sequence databases. This is achieved by considering both halves of basepaired positions simultaneously when scoring a sequence against a profile. For example, in Figure 1, positions 3 and 8 form a basepair in the consensus structure. These two positions are completely variable at the sequence level and so contribute

zero bits of information to a sequence profile. However, of the $4x4 = 16$ possible basepairs at these positions, only the four possible Watson-Crick basepairs (A:U, U:A, C:G, G:C) exist in the homologs. By specifying that a match to the profile must contain a Watson-Crick basepair at these positions we've gained two additional bits of information (by reducing 16 possibilities to 4, corresponding again to answering two yes/no questions). Importantly, modeling structure will only add information in cases where the sequence varies and paired positions covary to maintain a basepair in the structure. For example, in Figure 1 positions 1 and 11 pair in the consensus structure, but are completely conserved in sequence, each contributing two bits to a sequence profile, and collectively contributing four bits to a sequence and structure profile (reducing 16 possibilities to 1), thus contributing the same amount of information to either a sequence-only or a sequence and structure profile.

For this example, we gain 3 additional bits of information from considering structure, decreasing our chances of finding a match in a random database by a factor of $2^3 = 8$, from once every $2^{14} = 4096$ nucleotides to once every $2^{17} = 32764$ nucleotides. For real functional RNAs the additional amount of information gained from modeling structure varies widely. Figure 2 shows the information in a sequence and structure profile (CM) versus a sequence-only profile (HMM) for about 100 RNA families. Some RNAs, like tRNA, include about as much information in their structure as in their sequence, while for others, the increase is relatively modest. Note that for most families, modeling structure contributes at least 10 additional bits of information, which corresponds to lowering the expected chance of a false positive in a random database (i.e. the E-value of a database hit) by three orders of magnitude ($2^{10} = 1024$).

1.2.1 CMs are probabilistic models

In the previous example, sequences were either matches or mismatches to a profile, a simple yes/no scheme that offers no information on how *good* a match is. As SCFGs, CMs are importantly different from this simple match/mismatch paradigm in that they assign probabilities to the alignment of each nucleotide of a target sequence to each position of the profile, instead of a binary yes/no decision for a match/mismatch. Additionally, in a CM, nucleotides can be inserted and deleted relative to the consensus model, corresponding to an alignment of a gap in the model to a nucleotide in the target, and a consensus nucleotide in the model to a gap in the target, respectively. To facilitate the handling of insertions and deletions, CMs are organized as a binary tree of states, with each single-stranded position or basepair of the consensus sequence and structure modeled by separate match, insert, and delete states, corresponding to a consensus match, insertion after, or deletion of the relevant position/pair in the model. The topology of the tree mirrors the branching pattern of the consensus structure. States are connected to a subset of other states by *transitions*, each associated with a probability, and nucleotides are emitted by (aligned to) match and insert states according to state-specific emission probabilities. CM states and transitions are equivalent to SCFG nonterminals and production rules.

Given a particular alignment and secondary structure, the CM grammar formalism unambiguously dictates the construction of a particular tree topology of states and possible transitions between those states. The emission and transition probabilities for each state are then defined as mean posterior estimates based on the observed counts in the input alignment position(s) modeled by the state and a mixture Dirichlet prior (for emissions) or single component Dirichlet prior (for transitions). The details of this construction and parameterization procedure are not particularly relevant here and so are omitted; for more information see [23, 15, 17, 58, 59].

1.2.2 Scoring sequences with the CM Inside and CYK algorithms

Given a fully parameterized model M and a target sequence s, CM implementations, such as Infernal, calculate a log-odds score that the sequence was generated by the CM versus by a background null model R. The null model typically used is a simple generative model of 25% A, C, G, and U, from which the probability of generating any sequence of length L is simply 0.25^L. This log-odds score is calculated by the CM Inside dynamic programming algorithm as:

$$S_{Inside} = \log_2 \frac{P(s|M)}{P(s|R)} = \sum\nolimits_{\pi} \frac{P(s,\pi|M)}{P(s|R)}, \tag{1}$$

where π is a particular state path (i.e. alignment to the model, equivalent to a SCFG parse tree) through M that could have generated sequence s. The CM CYK dynamic programming [15] algorithm calculates a similar score: the log-odds score that s was generated by the maximum likelihood state path $\hat{\pi}$ that could have generated s, versus the same null model R. Specifically CYK calculates:

$$S_{CYK} = \log_2 \frac{P(s,\hat{\pi}|M)}{P(s|R)}. \tag{2}$$

The Inside score is more appropriate for determining if a sequence is homologous based on the model because it effectively integrates out the nuisance variable of the state path of the sequence in question. However, the CYK algorithm is also useful in practice because, due to details of their implementations in Infernal, CYK is about three times faster than Inside, and the CYK score approximates the Inside score well for most high scoring sequences of interest (because a single path accounts for a large fraction of the total probability mass of all paths). To accelerate searches, Infernal uses CYK as a filter for Inside, as explained later.

As presented above, the Inside and CYK algorithms compute log-odds scores for a complete target sequence s, but in practice RNA homologs are relatively short regions within long sequences. Infernal implements variants of Inside and CYK that scan along a target sequence scoring all possible subsequences as potential homologs. Because the log-odds scores are of base 2, the scores are in units of bits. An Inside score of x bits for a target sequence means that the sequence was $y = 2^x$ times more likely to have been generated by the CM than by the background model; for $x = 10$, y is 1024, and for $x = 20$, y is $1,048,576$.

CM search algorithms are computationally expensive. Empirically, CYK scales $O(LN^{2.4})$ for a model of N consensus positions and a database of length L [58]. Inside has the same asymptotic time complexity as CYK, but is roughly three times slower in practice. Search times with the standard CYK and Inside algorithms are often impractically slow. For example, to search a typical sized archaeal genome (about 6 million bases (Mb), two strands of a 3 Mb genome) with a tRNA model of 71 consensus positions and a SRP RNA model of 302 consensus positions using the Inside algorithm requires about 2.5 and 21 CPU-hours respectively. To repeat the same searches on the 3 Gb chimpanzee genome requires 0.3 and 2.5 CPU-years.

In contrast, the profile HMM Viterbi and Forward algorithms, which are analogous to CYK and Inside, scale $O(LN)$ [15]. Consequently, HMM searches take far less time than CM searches, especially for large models. For example, searching the chimpanzee genome with profile HMM models of tRNA and SRP using Infernal version 1.0's implementation of the Forward algorithm require 8 and 30 CPU-hours respectively, making them about 300 times and 700 times faster than the CM Inside algorithm. More recent implementations of profile HMM algorithms are even faster. The HMMER3 software package uses heuristic filters to rapidly remove the majority of the database quickly and only applies the Forward algorithm to the surviving fraction, resulting in 100- to 1000-fold acceleration for profile HMM searches for protein families at a negligible cost to sensitivity [21, 22].

1.3 Infernal

Infernal is a software package that implements CM methods. It includes programs to build a CM from an alignment (cmbuild), search a target sequence database with a CM (cmsearch) and create multiple sequence alignments of putative homologs with a CM (cmalign). Additionally, models are "calibrated" with the cmcalibrate program prior to using cmsearch. Calibration enables the reporting of expectation values (E-values) for putative homologs found in database searches. Infernal is an updated version of the Cove software package [16] which is used by tRNAscan-SE and SRPscan.

To alleviate slow search speeds, the latest version of Infernal (v1.1) executes multiple rounds of filtering of the target database prior to using Inside, the slowest but most sensitive CM search algorithm. The earliest rounds of the filter pipeline use a profile HMM to rapidly scan each target sequence and identify subsequences that may contain high-scoring hits to the CM based on sequence conservation alone. These filters are very similar to those employed in the HMMER3 pipeline [22], albeit with different survival thresholds such that a larger fraction of the database is expected to survive. The relaxed thresholds are important to ensure that hits with low sequence similarity but high structural similarity to the model will survive to the downstream CM stages of the pipeline. Subsequences that survive the profile HMM filters are then scored with a constrained version of the CM CYK algorithm. The CYK constraints are derived from a profile HMM alignment of the target sub-

sequence, and limit the range of positions of the subsequence that are permitted to align to each state of the CM. These constraints are enforced as bands on the CYK dynamic programming matrix and result in a significant acceleration versus standard, non-banded CYK, especially for large RNAs (often up to or exceeding 100-fold acceleration) [9, 57]. Subsequences surviving the HMM banded CYK filter are evaluated with the Inside algorithm, again using HMM-derived bands, to assign their final scores. For more details on Infernal's filter pipeline see [59]. The pipeline accelerates typical CM searches by three to four orders of magnitude versus non-filtered, non-banded Inside-only searches at a small cost to sensitivity, and enables CM searches of large genomes in a reasonable amount of time.

Parallelization is another strategy Infernal uses for decreasing running times when a compute cluster is available. The cmalign, cmsearch and cmcalibrate programs are implemented in coarse-grained parallel MPI versions allowing, for example, a search of a large vertebrate genome to finish faster by spreading the search across multiple nodes of a cluster.

1.4 Rfam

Rfam is a database of RNA families, each represented by a CM and two different multiple sequence alignments called a *seed* and a *full* alignment [28]. The seed alignment is a manually curated alignment of representative members of the family that is used to construct a CM using Infernal's cmbuild program. The CM is then searched against a large sequence database called RFAMSEQ based on a particular release of EMBL [47]. For each family, an Rfam curator chooses a bit score threshold, called the gathering threshold (GA), that separates the first clear false positive from trusted true homologs. All hits with bit scores above this threshold are extracted and aligned to the model to create the full alignment, which is not refined further. The most current release of Rfam (10.1) includes 1973 RNA families and annotates 2,756,313 regions in the 170 Gb RFAMSEQ database, each of which was scored by a model above its GA threshold and is included in a full alignment. Notably, the CMs provided by Rfam come pre-calibrated and so will report E-values when used by cmsearch.

2 Using Infernal to annotate structural RNAs in an archaeal genome

In this section, I'll guide the reader through an exercise of using Infernal and Rfam to annotate functional RNAs in the genome of *Methanobrevibacter ruminantium* (GenBank accession CP001719.1), a methanogenic archaeon that lives in the stomachs of ruminant mammals such as cows [46]. This particular archaeon was chosen because the analysis of the search results illustrate some important considerations

regarding the Infernal/Rfam strategy for genome annotation of RNAs, as discussed later in section 2.1. For this exercise, it is assumed that the reader is familiar with a command-line Unix environment and has some experience writing simple scripts. The specific instructions here correspond to release 10.1 of the Rfam database and version 1.1 of Infernal. If you are using a more recent version of Rfam than 11.0 you should follow slightly different instructions; see the "Notes" section at the end of this chapter.

Step 1. Download and install Infernal 1.1.
Step 2. Download the 102 CMs from Rfam 10.1 that match at least one archaeal sequence from RFAMSEQ.
Step 3. Convert the Infernal 1.0 Rfam CM file to Infernal 1.1 format.
Step 4. Download the *M. ruminantium* genome sequence from NCBI.
Step 5. Run CM searches against the genome.
Step 6. Analyze the results.

Step 1. Download and install Infernal 1.1.

Go to *http://infernal.janelia.org/* and download version 1.1 of Infernal, then unzip and untar it. The user's guide will be in *infernal-1.1/Userguide.pdf*, which contains installation instructions. For a basic installation, simply execute *./configure; make* from the infernal-1.1 directory. This will create Infernal executable files in the *infernal-1.1/src/* directory, for example the programs *cmsearch* and *cmconvert*, which we'll use here. For steps 3 and 5 to work, you'll need to make sure that these programs are in your path (so that when you type *cmsearch* it executes the *cmsearch* program you just built). To install these programs in system-wide directories, execute *make install*. See the user's guide for more information on installation. (At the time of writing, the most current available version of Infernal is actually 1.1rc1, the first *release candidate* for version 1.1. As you read this it is likely that the final version 1.1 will be available, or perhaps even a newer version. Note that the results here may only be exactly reproducible using version Infernal version 1.1rc1 and Rfam release 10.1.)

Step 2. Download the 102 CMs from Rfam 10.1 that match at least one archaeal sequence from RFAMSEQ.

Go to *http://rfam.sanger.ac.uk/* and click on *Taxonomy Search* on the left hand side of the page and search for *archaea*. The next page should report that 102 families were found.

Next, download the 102 CMs from Rfam. At the time of writing, users can either download all 1973 CMs in the database in a single file, or one at a time each in a separate file. The easiest option is probably to download all the models and then write a script to select the desired 102. To do this, create a text file called *arc.102.list*

and copy the names of the 102 families into it. Then, from *http://rfam.sanger.ac.uk/* click on *FTP* at the top of the page, and click on *CURRENT* and download the file *Rfam.cm.gz*.

Create a new directory and place the files *arc.102.list* and *Rfam.cm.gz* in it and decompress *Rfam.cm.gz* with *gunzip*. Next, you'll have to write a script to extract the 102 CMs that are listed in *arc.102.list* from the *Rfam.cm* file. There are many ways to do this. There are two important aspects of the CM file format that you'll need to know about. The first is that the *Rfam.cm* file is a concatenation of 1973 individual CM files, each beginning with a line that reads *INFERNAL-1 [1.0]*, , and ending with the line //. Secondly, the name of the model appears immediately after the *INFERNAL-1 [1.0]* line. The extraction script will need to read through the file, printing out only those lines from the models listed in *arc.102.list*. Save the 102 models as the file *arc.102.old.cm*.

Step 3. Convert the Infernal 1.0 Rfam CM file to Infernal 1.1 format.

Perform the conversion using Infernal 1.1's *cmconvert* program, with the command:

```
cmconvert arc.102.old.cm > arc.102.cm
```

The conversion should take about three minutes.

Step 4. Download the *Methanobrevibacter ruminantium* genome sequence from NCBI.

Go to the ENTREZ search page: *http://www.ncbi.nlm.nih.gov/sites/gquery*, search for *CP001719.1* and follow the link for the *Nucleotide* database. A page should load reading: *Methanobrevibacter ruminantium M1 chromosome, complete genome*. To download the genome in FASTA format, click on the *Send* link on the right hand side of the page and select *Complete Record*, *File* and FASTA format; then click *Create file*. A file called *sequences.fasta* should download. Rename this file *mrum.fa* and move it to the directory where you've stored the converted CM files from step 3.

Step 5. Run CM searches against the genome.

Now you are ready to search the *M. ruminantium* genome with the Rfam CMs using Infernal. CM searches are computationally expensive, but not impractical. All 102 searches should take less than ten minutes, and can be executed with a single command:

```
cmsearch --cut_ga --tblout mrum.tbl arc.102.cm mrum.fa > mrum.cmsearch
```

This command includes two options. The `--cut_ga` option tells the program to set the score threshold for reporting a hit to each model as that model's manually curated Rfam gathering threshold. The `--tblout mrum.tbl` option specifies

that a tabular version of the search results be printed to a file called *mrum.tbl* as explained below in step 6.

Step 6. Analyze the results

When the search finishes running you should have two new files in your directory called *mrum.cmsearch* and *mrum.tbl* containing the search results. The former file includes information on the high-scoring regions or *hits* in the genome to each model, including the bit scores and E-values of those hits as well as alignments of each of the hits to its respective model. The latter file contains much of the same information but in a simplified one-line-per-hit format which can be easily parsed by scripts one might use to analyze the results. Next, we'll take a closer look at example results from each of these files.

Open the *mrum.tbl* file and look at the first few lines (these have been split in half below to fit on the page):

```
#target name                accession query name           accession mdl mdl from   mdl to seq from ...
#--------------------------- --------- -------------------- --------- --- -------- -------- -------- ...
gi|288541968|gb|CP001719.1| -         5S_rRNA              -          cm        1      119   766016 ...


  ...  seq to strand trunc pass   gc  bias  score   E-value inc description of target
  ... ------- ------ ----- ---- ---- ----- ------ --------- --- ---------------------
  ...  766137      +    no    1 0.48   4.0   58.9   1.4e-11 !   Methanobrevibacter ruminantium M1, comp...
```

As indicated by the column names, this line reports a hit from position 766016 to 766137 of the genome (with target sequence name gi|288541968|gb|CP001719.1| in the *mrum.fa* file) to the 5S_rRNA Rfam CM, with a bit score of 58.9 bits and an E-value of $1.4e-11$. This E-value indicates that the probability of finding a hit with this bit score or higher is about 10^{-11} in a database the size of this genome. Because this E-value is so low, we can be confident that this region is indeed a homolog of 5S rRNA. As discussed later, annotators can be fairly confident of hits with E-values up to about $1e-5$ in archael and bacterial genomes. Additionally, we know that this hit has a bit score that is at least as high as the Rfam gathering (GA) threshold for the 5S_rRNA model. In fact, *all* the hits in our search results will meet or exceed the GA threshold for their respective models because we chose the `--cut_ga` option when running cmsearch. In addition to considering the E-value and bit score of a hit, inspection of the alignment of the hit to the query CM is often useful when determining if a hit is a real homolog or not. Alignments are not contained in the tabular output files, but are included in the standard cmsearch output. To find the alignment of this hit to the tRNA model in the *mrum.cmsearch* file, it is useful to know that the file is organized into 102 sections, one for each query CM. Each CM's section begins with "Query:" at the beginning of a line followed by the CM name, and then a list of hits ranked by E-value (lowest to highest), and then the hit alignments for all reported hits in the same ranked order.

The first tRNA hit alignment in *mrum.cmsearch* is for the 69.8 bit hit from positions 735136 to 735208 of the *M. ruminantium* genome, target sequence gi|288541968|gb|CP001719.1|, organized into a single block of six lines, shown in Figure 3. (Longer alignments, such as the second tRNA alignment in this file, will

be split into multiple blocks of six lines each.) The second line of each block ends with `CS` and shows the secondary structure of the consensus tRNA molecule modeled by the CM. This structure is shown as the leftmost secondary structure at the bottom of Figure 3. In the alignment, positions of the model that are paired are indicated by either parentheses or brackets (i.e. `(),<>`) and the left and right half of pairs are identified by matching left and right parentheses or brackets from outside to inside as in a mathematical formula. For example, the first (leftmost) "`(`" matches the last (rightmost) "`)`" indicating that these two positions are basepaired with each other. Similarly the second "`(`" matches the second to last "`)`", and so on. The difference between parentheses and brackets indicates levels of nesting. For example, the parentheses depict the acceptor stem between the 5' and 3' ends of the tRNA (the top stem in the structures in Figure 3), while the three other stems are indicated by brackets because they are independent stems and are fully contained between the two halves of the acceptor stem. Other RNAs in Rfam, such as SSU_rRNA_archaea and RNase_P_arch, have more than two nesting levels of stems in their consensus secondary structures, and to handle these cases additional characters are used (i.e. `{},[]`).

The first line of the alignment ends with `NC`. This line indicates negative scoring noncanonical basepairs, these are basepairs in the target sequence which receive a negative score to the model and are not either Watson Crick (A:U, U:A, C:G, G:C) or wobble (G:U, U:G) basepairs. A negative score is assigned to basepairs that are less probable in their particular position of the CM than in the random background model, i.e. less than $1/16 = 0.0625$. There are zero such basepairs in this target sequence, so this line is entirely blank. If any such basepairs did exist (there are some examples in other alignments in this file) they would be highlighted with a `v` character in this line.

The third line of the alignment shows the query model consensus sequence. This is defined as the highest scoring nucleotide or basepair at each position, with capital letters being highly conserved and lowercase letters being less well conserved. Dots in this line indicate a position where the target sequence has inserted one or more residues. The fifth line shows the target sequence, in this case ranging from position 735136 to 735208. Lowercase nucleotides here, such as the single lowercase `c` in this line, indicate inserted nucleotides relative to the consensus model. The fourth line in each block indicates how well the query and target align to each other and is meant to help the user quickly judge the quality of the alignment when examining a putative homolog. If a nucleotide *N* is present, then the target has the most probable nucleotide *N* aligned at that position. If a blank space or non-alphabetic character is present, then the target contains either a gap or a nucleotide other than the most probable one. Of these cases, a "`+`" or "`:`" occurs when the target nucleotide receives a positive score for the model, either for single stranded positions (`+`) or basepaired positions (`:`). A blank space occurs if either the target nucleotide is a gap or if the target nucleotide receives a negative score to the model. As explained earlier, a positive score indicates the nucleotide or basepair is more probable than the random background model (i.e. has higher probability than 0.25 or 0.0625, respectively).

Finally, the sixth line ends in `PP` and indicates the confidence level, or expected accuracy, of each position of the alignment. Each position receives a single character summarizing it's posterior probability. A `0` means 0-5%, a `1` means 5-15%, and so on; a `9` means 85-95%, and a `*` means 95-100% posterior probability. In this alignment all positions are `*` indicating they are all very confidently aligned correctly, but there are examples of more ambiguous alignments elsewhere in this file. As you might expect, alignment positions with low confidence are often nearby insertions and deletions.

2.1 Important considerations regarding Infernal predictions

Table 4 reports the number of hits found in the *M. ruminantium* genome with scores that exceed the Rfam gathering threshold for each of the 102 families with at least 1 such hit. There are 128 total predicted RNAs from 8 different Rfam families (after removing overlaps and hits with marginal E-values, as explained more below), as opposed to only 66 from 4 different families in the NCBI GenBank and Refseq annotation (accessions CP001719.1 and NC_013790.1). Closer scrutiny of these results offer insights into some important strengths, weaknesses and caveats of using Infernal and Rfam for genome annotation. Below, these issues are listed and explained using specific examples from the results.

1. Hits from different models can overlap.

It is not uncommon for a single region of a genome or target database to be hit by multiple models from Rfam. There are several reasons why this may occur. First, some families are evolutionarily related to each other. An example of this in the *M. ruminantium* results are the SSU_rRNA_archaea and SSU_rRNA_bacteria models, which model archaeal SSU ribosomal RNA and bacterial SSU ribosomal RNA respectively. The SSU rRNA is ancient and predates the split of the three domains, so a homology search method that identifies cross-matches between these families is correct. Note that, as expected, the score for the hit to the archaeal model is much higher than the bacterial model (1483.0 versus 1090.7). Another example are the overlaps between nearly all of the roughly sixty CRISPR-DR2 and CRISPR-DR39 hits. Both of these CRISPR CMs have 30 consensus positions, and they share some sequence and structural similarity. When overlapping hits are encountered, it is recommended practice to keep the hit with the better E-value. In most cases this will be the hit with the higher bit score, but not always. If both hits have the same E-value (as with the two hits of E-value 0 to the SSU models), keep the one with the higher bit score.

2. Hits with marginally significant E-values should be carefully examined or thrown out.

To avoid misannotating RNAs in a genome, the E-values of predicted hits should be considered, even for hits above the Rfam GA bit score threshold. In these searches, use of the cmsearch `--cut_ga` option dictated that only hits exceeding the GA threshold be reported. Because we searched with 102 models, the highest-scoring false positive hit to any family we expect is about $1/102$, which is roughly 0.01. However, because we also only consider hits above the GA bit score thresholds and for some families those thresholds correspond to E-values below 0.01, this calculation is only roughly accurate. In general, when performing N searches, the highest-scoring false positive hit should have an E-value of roughly $1/N$ because that E-value literally means that we expect $1/N$ such hits from a particular search, so $N * 1/N = 1$ such hits are expected in N searches. As the predicted E-value of the highest-scoring false positive, 0.01 is a reasonable E-value threshold to use during annotation. As shown by the differences between the "believed" and "unique" columns of Table 4, doing so in this analysis would lead to the removal of the following unique (nonoverlapping) hits: the single hit to the sR11 model and the two hits to the CRISPR-DR39 model because these three hits have E-values above 0.01. One additional "unique" hit is not counted in the "believed" column: the CRISPR-DR42 model with an E-value of 0.0045. This E-value is slightly below the 0.01 threshold but is ruled out because CRISPR sequences almost always occur as tandem repeats as explained in the next section.

In fact, because Infernal E-values are not perfectly accurate, even more conservative E-value thresholds are often used in practice. For example, only hits with E-values below $1e-5$ were considered in a recent survey of SmY RNAs in nematodes [40]. In our results, dropping the E-value threshold from 0.01 to $1e-5$ would additionally exclude only the 60 remaining CRISPR-DR2 hits. However, as discussed next, closer scrutiny of these CRISPR-DR2 predictions suggests they are in fact real homologs. Even stricter E-values may be necessary for searches of complex genomes which strongly invalidate the assumptions made by the Infernal E-value machinery. For example, large vertebrate genomes that include high numbers of tandem repeats can pose particular problems for Infernal, as discussed in point 5 below.

3. Expert knowledge of a family can help verify an Infernal prediction.

Often it is possible to gain corroborating evidence that an Infernal prediction is either a true homolog or not based on additional knowledge of the family that cannot be modeled by a CM. For example, CRISPR genes are Clustered Regularly Interspaced Short Palindromic Repeats that are separated by spacers of similar length. The Infernal CRISPR-DR2 predictions follow this pattern: 60 out of 61 are identical 30 nucleotide subsequences, and 59 of those 60 are separated by between 91 and 98 nucleotides (the remaining spacer is 160 nucleotides). In contrast, the

61st hit to this model occurs about 500 Kb away from the cluster of 60, and has a marginal E-value of 0.11 suggesting it is probably not a real CRISPR element. By the same reasoning, the single hit to the CRISPR-DR42 model is likely a false positive hit even though it has a more significant E-value of 0.0045.

Another example involves the single highly significant (E=$8.8e-28$) hit to the FMN riboswitch model. Because riboswitches tend to occur in the 5' untranslated region of genes involved in metabolism of a particular ligand, flavin mononucleotide (FMN) for this particular switch, additional evidence that a predicted riboswitch is real is often obtained by examining the function of downstream protein-coding genes. In this case, the nearest gene is annotated as a *ribB* gene, a 3,4-dihydroxy-2-butanone 4-phosphate synthase, which is involved in riboflavin metabolism, and importantly, the predicted riboswitch is on the same strand and is 5' of the coding sequence of *ribB*. These data suggest the Infernal prediction is in fact an FMN riboswitch.

4. Some RNA families are not included in Rfam, others are represented by models that are not full-length.

There are two common reasons for a family's absence from Rfam. First, it may have just been discovered. Novel families continue to be discovered at a rapid pace [73, 74, 40, 75] making it difficult for the limited number of Rfam curators to incorporate all of them into the database. Secondly, some RNAs are so large that running the Rfam search pipeline for them would take an impractical amount of compute time due to the high complexity of the CM Inside and CYK algorithms. A glaring omission from Rfam that falls into the second category is LSU rRNA models. However, as mentioned earlier, LSU is highly conserved at the sequence level and using CM methods to identify them is unnecessary because more efficient sequence-based methods can do the job well, such as the profile HMM approach taken by the RNAmmer program [42]. In the future, Rfam could take a similar approach and use profile HMMs for LSU searches. However, currently an Infernal user aiming for a complete annotation of RNAs in a genome would need to run an additional program such as RNAmmer or BLASTN to annotate full length LSU sequences.

Some families in Rfam are represented by alignments and models that do not cover the full length of the sequence family. Two examples are the group I and group II self-splicing intron models. These models are not full length because the high variability in the sequence and structure of homologs makes construction of a reasonable global structural alignment difficult. In such cases, only the well-conserved core regions are modeled, and the variable parts are omitted. While largely incomplete models like these are rare, nearly-complete models that may not include the complete 5' and 3' ends of the RNA are more common. One reason for this is that the structures and alignments for some Rfam families are based mainly on predictions from comparative sequence analysis and experiments to determine the precise start and end points of the non-coding transcripts have not yet been performed.

5. Eukaryotic sequences offer additional challenges.

For Infernal searches of eukaryotic genomes, new issues arise and some of the problems discussed above can become more severe. Eukaryotic genomes contain certain types of sequence elements largely absent from archaea and bacteria that can lead to high-scoring false positives in CM searches, namely pseudogenes and repeats. For example, some short interspersed nuclear elements (SINEs) are derived from pol-III transcribed RNAs like tRNA or SRP RNA. Examples include Alu sequences, which are common in primates, numbering greater than 1 million in the human genome. Pseudogenes of U6, 7SK and Y RNAs are also common [32]. These elements will often score high to a CM due to their homology with the original RNA family from which they were derived. An example of this problem is shown in Table 6, which contains results of Infernal searches and family-specific searches for selected families in four eukaryotic genomes (described more in the next section). The table shows that thousands of tRNA-derived SINEs are identified by tRNAscan-SE in the mouse genome (*Mus musculus*). These elements were noted in the original publication for that genome [72]. tRNAscan-SE identifies 26,201 regions in the genome, 22,918 of which are reported as pseudogenes, and 3,283 of which are predicted as tRNA genes. All but about 500 of these were discounted after closer inspection as likely non-functional SINE repeats in [72].

Additionally, less complex inverted tandem repeats often score high against any CM with a single stem loop (such as miRNAs) or otherwise simple secondary structure because opportunities for stretches of Watson-Crick basepairing between nearby regions in these elements are abundant. Large numbers of high-scoring false positives greatly complicates the analysis of Infernal results because while it is desirable to set a single E-value threshold for all families, in reality, certain families will require special treatment.

2.2 Comparison of Infernal to family-specific methods

In the *M. ruminantium* searches, Infernal was able to improve upon the existing RNA annotation by finding two probable tRNAs missed by tRNAscan-SE (Figure 4), suggesting that Infernal can be more sensitive than family-specific methods in some cases. For further comparison, I used some popular family-specific methods and the corresponding Rfam CMs with Infernal to search the fourteen genomes listed in Table 1. A comparison of the results is shown in Tables 4 (archaeal genomes), 5 (bacterial genomes), and 6 (eukaryotic genomes).

The Infernal results largely agree with the tRNAscan-SE, SRPscan and Bcheck results. This is not surprising considering that all of these programs, including Infernal, are using CMs with sequence-based filters. The main difference is in the design of those filters. For Infernal, profile HMMs built from the CM are used, whereas for the others, sequence and structural characteristics of the specific families being modeled have been exploited to enable stricter filtering in some cases.

The stricter filtering can enable faster searches in some cases (e.g. Aragorn searches for tmRNAs in bacteria) but can also cause high-scoring hits to be missed, such as Bcheck's failure to identify any RNaseP RNAs in *M. ruminantium* (Table 4). Additional examples are the SRP RNA prediction by Infernal in *M. ruminantium* which is missed by SRPscan, and the three archaeal tRNAs predicted by Infernal but not by tRNAscan-SE, two of which are shown in Figure 4. Conversely, because Infernal's HMM filters do not consider structure they could miss some high-scoring hits that the other methods find. In these searches, this is exemplified by putative tRNAs predicted by Aragorn and tRNAscan-SE that are not found by Infernal. However, with the exception of tRNA for several genomes, and 5S and SSU rRNA in *A. thaliana*, Infernal finds all of the hits that the family-specific methods report.

Besides being faster in some cases, family-specific tools offer some other important advantages over Infernal, such as offering additional information relevant to the annotations. For example, tRNAscan-SE reports on the tRNA type in its predictions based on the anticodon sequence, as well as whether the tRNA contains an intron or is a predicted pseudogene. Also, some of family-specific CM based tools include more CMs than are present in Rfam, and the models are built from more carefully curated input alignments than those in Rfam in some cases. For example, Bcheck includes two archaeal RNase P CMs, while Rfam includes only one. Using more and/or better models can lead to more accurate or more complete annotations.

The comparison of RNAmmer with Infernal highlights an important difference between profile HMMs (as implemented in RNAmmer) and CMs. While most of the predictions agree, RNAmmer failed to recognize some 5S rRNA candidates in archaea that Infernal finds. This suggests that the additional statistical power gained by modeling the conserved 5S rRNA secondary structure is critical for the CM in these cases.

There are other RNA search tools that have not been tested here for various reasons. SnoReport[36] requires candidate RNA sequences as input and cannot scan along genome length sequences. Snoscan [52] and snoGPS [66] which identify C/D box snoRNAs and H/ACA box snoRNAs respectively and take advantage of user-specified ribosomal RNA sequences that include potential methylation/pseudouridylation sites for the predicted snoRNAs, complicating a potential comparison with a general CM approach. The Riboswitch finder [7] and RibEx [1] tools are only available via webservers that do not accept genome length sequences. The microRNA detection program RNAmicro [37] was not tested because it requires an alignment of orthologous sequences as input. Pattern search tools, such as RNA-motif [53], and RNABOB [18] were not tested because libraries of patterns analogous to Rfam CMs that would enable analogous searches are not readily available. Other tools, such as RNA-PATTERN [41], are not freely available for download.

3 Conclusion

As demonstrated here by the example analysis of the *M. ruminantium* genome, using Infernal and Rfam instead of more commonly used tools can lead to a more complete annotation of RNAs in genomes. The Infernal results contain important information on the biology of *M. ruminantium* that is absent from its initial GenBank annotation. For example, the existence of sixty CRISPR hits indicates that *M. ruminantium* can likely acquire resistance against viruses through the CRISPR system [6]. Additionally, the presence of a high-scoring hit (110 bits, E-value of $8.8e-28$) to the FMN riboswitch model strongly indicates that this archaeon encodes a true riboswitch which may control expression of at least some genes involved in riboflavin biosynthesis through binding of FMN to this structural element. This is especially interesting because riboswitches primarily exist in bacteria. Further, a single RNase P RNA and a single SRP RNA have been predicted, which is expected but still relevant because these RNAs were not annotated in the initial publication of the genome.

For annotation of functional RNAs in genomes, the general Infernal/Rfam approach is comparable in both speed and sensitivity to the use of family-specific tools that utilize specialized filters or search algorithms. The total time required for the 102 *M. ruminantium* searches was about 6 minutes on a single CPU. For families for which specific search tools exist, their results largely agree with Infernal (Tables 4, 5, and 6). Importantly though, the Rfam database includes a growing number of CMs of families for which specific search tools capable of scanning genomes do not exist, which can be used by Infernal to search for as-of-yet undiscovered homologs. Infernal has the added advantage of convenience for annotation pipeline developers: it is a single program that works for most RNA families, making it easier to incorporate and maintain in a pipeline than multiple family-specific programs.

4 Notes

If you are working through the *M. ruminantium* genome annotation example in section 2 and have access to an Rfam release more recent than 11.0 that was based on Infernal 1.1, you should be able to simplify the six step annotation process. The first important change is that you do not need to write a script to extract the archaeal CMs from the *Rfam.cm* file. Instead, use Infernal's *cmfetch* program (see the Infernal user's guide for details). Secondly, you can skip step 3, the CM conversion step. The other steps should be followed as written in section 2 but your results will likely be slightly different. For example, you will probably be working with more than 102 CMs.

Acknowledgements I thank Sean Eddy and Tom Jones for useful discussions and critical comments on the manuscript.

References

[1] C. Abreu-Goodger and E. Merino. RibEx: a web server for locating riboswitches and other conserved bacterial regulatory elements. *Nucleic Acids Res*, 33:W690–W692, 2005.

[2] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. 25:3389–3402, 1997.

[3] S. Altuvia, A. Zhang, L. Argaman, A. Tiwari, and G. Storz. The *Escherichia coli* OxyS regulatory RNA represses FhlA translation by blocking ribosome binding. *EMBO J*, 17:6069–6075, 1998.

[4] L. Argaman, R. Hershberg, J. Vogel, G. Bejerano, E. G. Wagner, H. Margalit, and S. Altuvia. Novel small RNA-encoding genes in the intergenic regions of *Escherichia coli*. *Curr. Biol.*, 11:941–950, 2001.

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

[6] R. Barrangou, C. Fremaux, H. Deveau, M. Richards, P. Boyaval, S. Moineau, D. A. Romero, and P. Horvath. CRISPR provides acquired resistance against viruses in prokaryotes. *Science*, 315:1709–1712, 2007.

[7] P. Bengert and T. Dandekar. Riboswitch finder–a tool for identification of riboswitch RNAs. *Nucleic Acids Res*, 32:W154–W159, 2004.

[8] L. Brocchieri and S. Karlin. Protein length in eukaryotic and prokaryotic proteomes. *Nucleic Acids Res*, 33:3390–3400, 2005.

[9] M. P. Brown. Small subunit ribosomal RNA modeling using stochastic context-free grammars. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 8:57–66, 2000.

[10] C. Burge and S. Karlin. Prediction of complete gene structures in human genomic DNA. 268:78–94, 1997.

[11] C. B. Burge, T. Tuschl, and P. A. Sharp. Splicing of precursors to mRNAs by the spliceosomes. In R. F. Gesteland, T. R. Cech, and J. F. Atkins, editors, *The RNA World, Second Edition*, pages 525–560. Cold Spring Harbor Laboratory Press, Cold Spring Harbor, New York, 1999.

[12] N. Bushati and S. Cohen. microRNA functions. *Annu Rev Cell Dev Biol.*, 23:175–205, 2007.

[13] A. Clum, B. J. Tindall, J. Sikorski, N. Ivanova, K. Mavrommatis, S. Lucas, T. Glavina, , M. Nolan, F. Chen, H. Tice, S. Pitluck, J. F. Cheng, O. Chertkov, T. Brettin, C. Han, J. C. Detter, C. Kuske, D. Bruce, L. Goodwin, G. Ovchinikova, A. Pati, N. Mikhailova, A. Chen, K. Palaniappan, M. Land, L. Hauser, Y. J. Chang, C. D. Jeffries, P. Chain, M. Rohde, M. Gker, J. Bristow, J. A. Eisen, V. Markowitz, P. Hugenholtz, N. C. Kyrpides, H. P. Klenk, and A. Lapidus. Complete genome sequence of *Pirellula staleyi* type strain (ATCC 27377). *Stand. Genomic Sci*, 1:308–316, 2009.

[14] A. L. Delcher, K. A. Bratke, E. C. Powers, and S. L. Salzberg. Identifying bacterial genes and endosymbiont DNA with Glimmer. *Bioinformatics*, 23:673–679, 2007.

[15] R. Durbin, S. R. Eddy, A. Krogh, and G. J. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge UK, 1998.

[16] S. R. Eddy. COVE - covariance models of RNA secondary structure. [ftp://selab.janelia.org/pub/software/cove/], 1996.

[17] S. R. Eddy. A memory-efficient dynamic programming algorithm for optimal alignment of a sequence to an RNA secondary structure. *BMC Bioinformatics*, 3:18, 2002.

[18] S. R. Eddy. RNABOB - fast pattern searching for RNA secondary structures. [ftp://selab.janelia.org/pub/software/rnabob/], 2005.

[19] S. R. Eddy. Computational analysis of RNAs. 71:117–128, 2006.

[20] S. R. Eddy. HMMER - biosequence analysis using profile hidden Markov models. [http://hmmer.janelia.org/], 2008.

[21] S. R. Eddy. A probabilistic model of local sequence alignment that simplifies statistical significance estimation. *PLoS Comput. Biol.*, 4:e1000069, 2008.

[22] S. R. Eddy. Accelerated profile HMM searches. *PLoS Comp. Biol.*, 7:e1002195, 2011.

[23] S. R. Eddy and R. Durbin. RNA sequence analysis using covariance models. 22:2079–2088, 1994.

[24] G. L. Eliceiri. Small nucleolar RNAs. *Cell Mol. Life Sci.*, 56:22–31, 1999.

[25] R. D. Finn, J. Mistry, J. Tate, P. Coggill, A. Heger, J. E. Pollington, O. L. Gavin, G. Ceric, K. Forslund, L. Holm, E. L. L. Sonnhammer, S. R. Eddy, and A. Bateman. The Pfam protein families database. 38:D211–D222, 2010.

[26] D. N. Frank and N. R. Pace. Ribonuclease P: Unity and diversity in a tRNA processing ribozyme. *Annu Rev Biochem.*, 67:153–180, 1998.

[27] E. K. Freyhult, J. P. Bollback, and P. P. Gardner. Exploring genomic dark matter: A critical assessment of the performance of homology search methods on noncoding RNA. *Genome Res.*, 17:117–125, 2007.

[28] P. P. Gardner, J. Daub, J. Tate, B. L. Moore, I. H. Osuch, S. Griffiths-Jones, R. D. Finn, E. P. Nawrocki, D. L. Kolbe, S. R. Eddy, and A. Bateman. Rfam: Wikipedia, clans and the "decimal" release. *Nucleic Acids Res*, 39:D141–D145, 2011.

[29] P. P. Gardner, J. Daub, J. G. Tate, E. P. Nawrocki, D. L. Kolbe, S. Lindgreen, A. C. Wilkinson, R. D. Finn, S. Griffiths-Jones, S. R. Eddy, and A. Bateman. Rfam: Updates to the RNA families database. 37:D136–D140, 2009.

[30] M. W. Gilmour, M. Graham, G. Van Domselaar, S. Tyler, H. Kent, K. M. Trout-Yakel, O. Larios, V. Allen, B. Lee, and C. Nadon. High-throughput genome sequencing of two *Listeria monocytogenes* clinical isolates during a large foodborne outbreak. *BMC Genomics*, 11:120, 2010.

[31] M. Gribskov, A. D. McLachlan, and D. Eisenberg. Profile analysis: Detection of distantly related proteins. 84:4355–4358, 1987.

[32] S. Griffiths-Jones. Annotating noncoding RNA genes. 8:279–298, 2007.

[33] R. Guig. Assembling genes from predicted exons in linear time with dynamic programming. *J Comput Biol*, 5:681–702, 1998.

[34] A. L. Hartman, C. Norais, J. H. Badger, S. Delmas, S. Haldenby, R. Madupu, J. Robinson, H. Khouri, Q. Ren, T. M. Lowe, J. Maupin-Furlow, M. Pohlschroder, C. Daniels, F. Pfeiffer, T. Allers, and J. A. Eisen. The complete genome sequence of *Haloferax volcanii* DS2, a model archaeon. *PLoS One*, 5:e9605, 2010.

[35] T. M. Henkin. Riboswitch RNAs: using RNA to sense cellular metabolism. *Genes Dev.*, 22:3383–3390, 2008.

[36] J. Hertel, I. L. Hofacker, and P. F. Stadler. SnoReport: computational identification of snoRNAs with unknown targets. *Bioinformatics*, 24:158–164, 2008.

[37] J. Hertel and P. F. Stadler. Hairpins in a haystack: Recognizing microRNA precursors in comparative genomics data. *Bioinformatics*, 22:e197–e202, 2006.

[38] P. Horvath and R. Barrangou. CRISPR/Cas, the immune system of bacteria and archaea. *Science*, 327:167–170, 2010.

[39] A. P. Jackson, J. A. Gamble, T. Yeomans, G. P. Moran, D. Saunders, D. Harris, M. Aslett, J. F. Barrell, G. Butler, F. Citiulo, D. C. Coleman, P. W. de Groot, T. J. Goodwin, M. A. Quail, J. McQuillan, C. A. Munro, A. Pain, R. T. Poulter, M. A. Rajandream, H. Renauld, M. J. Spiering, A. Tivey, N. A. Gow, B. Barrell, D. J. Sullivan, and M. Berriman. Comparative genomics of the fungal pathogens *Candida dubliniensis* and *Candida albicans*. *Genome Res*, 19(12):2231–44, Dec 2009.

[40] T. A. Jones, W. Otto, M. Marz, S. R. Eddy, and P. F. Stadler. A survey of nematode SmY RNAs. *RNA Biol.*, 6:5–8, 2009.

[41] M. D. Kazanov, A. G. Vitreschak, and M. S. Gelfand. Abundance and functional diversity of riboswitches in microbial communities. *BMC Genomics*, 8:347, 2007.

[42] K. Lagesen, P. Hallin, E. A. Rødland, H. H. Staerfeldt, T. Rognes, and D. W. Ussery. RNAmmer: consistent and rapid annotation of ribosomal RNA genes. *Nucleic Acids Res*, 35:3100–3108, 2007.

[43] D. Laslett and B. Cänback. ARAGORN, a program to detect tRNA genes and tmRNA genes in nucleotide sequences. 32:11–16, 2004.

[44] D. Laslett and B. Cänback. ARWEN: a program to detect tRNA genes in metazoan mitochondrial nucleotide sequences. *Bioinformatics*, 24:172–175, 2008.

[45] D. Laslett, B. Canback, and S. Andersson. BRUCE: a program for the detection of transfer-messenger RNA genes in nucleotide sequences. *Nucleic Acids Res*, 30:3449–3453, 2002.

[46] S. C. Leahy, W. J. Kelly, E. Altermann, R. S. Ronimus, C. J. Yeoman, D. M. Pacheco, D. Li, Z. Kong, S. McTavish, C. Sang, S. C. Lambie, P. H. Janssen, D. Dey, and G. T. Attwood. The genome sequence of the rumen methanogen *Methanobrevibacter ruminantium* reveals new possibilities for controlling ruminant methane emissions. *PLoS One*, 5:e8926, 2010.

[47] R. Leinonen, R. Akhtar, E. Birney, J. Bonfield, L. Bower, M. Corbett, Y. Cheng, F. Demiralp, N. Faruque, N. Goodgame, R. Gibson, G. Hoad, C. Hunter, M. Jang, S. Leonard, Q. Lin, R. Lopez, M. Maguire, H. McWilliam, S. Plaister, R. Radhakrishnan, S. Sobhany, G. Slater, P. Ten Hoopen,

F. Valentin, R. Vaughan, V. Zalunin, D. Zerbino, and G. Cochrane. Improvements to services at the *E*uropean *N*ucleotide *A*rchive. *Nucleic Acids Res*, 38:D39–D45, 2010.

[48] R. Lewin. Surprising discovery with a small RNA. *Science*, 218:777–778, 1982.

[49] H. Liesegang, A. K. Kaster, A. Wiezer, M. Goenrich, A. Wollherr, H. Seedorf, G. Gottschalk, and R. K. Thauer. Complete genome sequence of emphMethanothermobacter marburgensis, a methanoarchaeon model organism. *J Bacteriol*, 192:5850–5851, 2010.

[50] A. Lomsadze, V. Ter-Hovhannisyan, Y. O. Chernoff, and M. Borodovsky. Gene identification in novel eukaryotic genomes by self-training algorithm. *Nucleic Acids Res*, 33:6494–6506, 2005.

[51] T. M. Lowe and S. R. Eddy. tRNAscan-SE: A program for improved detection of transfer RNA genes in genomic sequence. 25:955–964, 1997.

[52] T. M. Lowe and S. R. Eddy. A computational screen for methylation guide snoRNAs in yeast. *Science*, 283:1168–1171, 1999.

[53] T. J. Macke, D. J. Ecker, R. R. Gutell, D. Gautheret, D. A. Case, and R. Sampath. RNAMotif, an RNA secondary structure definition and search algorithm. *NAR*, 29:4724–4735, 2001.

[54] A. V. Mardanov, V. A. Svetlitchnyi, A. V. Beletsky, M. I. Prokofeva, E. A. Bonch-Osmolovskaya, N. V. Ravin, and K. G. Skryabin. The genome sequence of the crenarchaeon *Acidilobus saccharovorans* supports a new order, Acidilobales, and suggests an important ecological role in terrestrial acidic hot springs. *Appl Environ Microbiol*, 76:5652–5657, 2010.

[55] G. Meister and T. Tuschl. Mechanisms of gene silencing by double-stranded RNA. *Nature*., 431:343–349, 2004.

[56] I. M. Meyer. A practical guide to the art of RNA gene prediction. *Brief. Bioinform*., 8:396–414, 2007.

[57] E. P. Nawrocki. *Structural RNA Homology Search and Alignment Using Covariance Models*. PhD thesis, Washington University School of Medicine, 2009.

[58] E. P. Nawrocki and S. R. Eddy. Query-dependent banding (QDB) for faster RNA similarity searches. *PLoS Comput. Biol*., 3:e56, 2007.

[59] E. P. Nawrocki and S. R. Eddy. The Infernal 1.1 user's guide. [http://infernal.janelia.org/], 2012.

[60] C. S. Peacock, K. Seeger, Dn Harris, L. Murphy, J. C. Ruiz, M. A. Quail, N. Peters, E. Adlem, A. Tivey, M. Aslett, A. Kerhornou, A. Ivens, A. Fraser, M. A. Rajandream, T. Carver, H. Norbertczak, T. Chillingworth, Z. Hance, K. Jagels, S. Moule, D. Ormond, S. Rutter, R. Squares, S. Whitehead, E. Rabbinowitsch, C. Arrowsmith, B. White, S. Thurston, F. Bringaud, S. L. Baldauf, A. Faulconbridge, D. Jeffares, D. P. Depledge, S. O. Oyola, J. D. Hilley, L. O. Brito, L. R. Tosi, B. Barrell, A. K. Cruz, J. C. Mottram, D. F. Smith, and M. Berriman. Comparative genomic analysis of three *Leishmania* species that cause diverse human disease. *Nat Genet*, 39(7):839–47, Jul 2007.

[61] W. R. Pearson. Effective protein sequence comparison. 266:227–258, 1996.

[62] N. K. Petty, R. Bulgin, V. F. Crepin, A. M. Cerdeo-Trraga, G. N. Schroeder, M. A. Quail, N. Lennard, C. Corton, A. Barron, L. Clark, A. L. Toribio, J. Parkhill, G. Dougan, G. Frankel, and N. R. Thomson. The *Citrobacter rodentium* genome sequence reveals convergent evolution with human pathogenic escherichia coli. *J Bacteriol*, 192:525–538, 2010.

[63] M. Regalia, M. A. Rosenblad, and T. Samuelsson. Prediction of signal recognition particle RNA genes. 30:3368–3377, 2002.

[64] S. W. Roh, Y. D. Nam, S. H. Nam, S. H. Choi, H. S. Park, and J. W. Bae. Complete genome sequence of *Halalkalicoccus jeotgali* B3(T), an extremely halophilic archaeon. *J Bacteriol*, 192:4528–4529, 2010.

[65] E. W. Sayers, T. Barrett, D. A. Benson, E. Bolton, S. H. Bryant, K. Canese, V. Chetvernin, D. M. Church, M. DiCuccio, S. Federhen, M. Feolo, I. M. Fingerman, L. Y. Geer, W. Helmberg, Y. Kapustin, D. Landsman, D. J. Lipman, Z. Lu, T. L. Madden, T. Madej, D. R. Maglott, A. Marchler-Bauer, V. Miller, I. Mizrachi, J. Ostell, A. Panchenko, L. Phan, K. D. Pruitt, G. D. Schuler, E. Sequeira, S. T. Sherry, M. Shumway, K. Sirotkin, D. Slotta, A. Souvorov, G. Starchenko, T. A. Tatusova, L. Wagner, Y. Wang, W. J. Wilbur, E. Yaschenko, and J. Ye. Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res*, 39:D38–D51, 2011.

[66] P. Schattner, W. A. Decatur, C. A. Davis, , M. J. Fournier, and T. M. Lowe. Genome-wide searching for pseudouridylation guide snoRNAs: analysis of the *Saccharomyces cerevisiae* genome. 32:4281–4296, 2004.

[67] R. L. Tatusov, M. Y. Galperin, D. A. Natale, and E. V. Koonin. The COG database: a tool for genome-scale analysis of protein functions and evolution. 28:33–36, 2000.

[68] A. Theologis, J. R. Ecker, C. J. Palm, N. A. Federspiel, S. Kaul, O. White, J. Alonso, H. Altafi, R. Araujo, C. L. Bowman, S. Y. Brooks, E. Buehler, A. Chan, Q. Chao, H. Chen, R. F. Cheuk, C. W. Chin, M. K. Chung, L. Conn, A. B. Conway, A. R. Conway, T. H. Creasy, K. Dewar, P. Dunn, P. Etgu, T. V. Feldblyum, J. Feng, B. Fong, C. Y. Fujii, J. E. Gill, A. D. Goldsmith, B. Haas, N. F. Hansen, B. Hughes, L. Huizar, J. L. Hunter, J. Jenkins, C. Johnson-Hopson, S. Khan, E. Khaykin, C. J. Kim, H. L. Koo, I. Kremenetskaia, D. B. Kurtz, A. Kwan, B. Lam, S. Langin-Hooper, A. Lee, J. M. Lee, C. A. Lenz, J. H. Li, Y. Li, X. Lin, S. X. Liu, Z. A. Liu, J. S. Luros, R. Maiti, A. Marziali, J. Militscher, M. Miranda, M. Nguyen, W. C. Nierman, B. I. Osborne, G. Pai, J. Peterson, P. K. Pham, M. Rizzo, T. Rooney, D. Rowley, H. Sakano, S. L. Salzberg, J. R. Schwartz, P. Shinn, A. M. Southwick, H. Sun, L. J. Tallon, G. Tambunga, M. J. Toriumi, C. D. Town, T. Utterback, S. Van Aken, M. Vaysberg, V. S. Vysotskaia, M. Walker, D. Wu, G. Yu, C. M. Fraser, J. C. Venter, and R. W. Davis. Sequence and analysis of chromosome 1 of the plant *Arabidopsis thaliana*. *Nature*, 408:816–820, 2000.

[69] H. J. Tripp, S. R. Bench, K. A. Turk, R. A. Foster, B. A. Desany, F. Niazi, J. P. Affourtit, and J. P. Zehr. Metabolic streamlining in an open-ocean nitrogen-fixing cyanobacterium. *Nature*, 464:90–94, 2010.

[70] M. Ventura, F. Turroni, A. Zomer, E. Foroni, V. Giubellini, F. Bottacini, C. Canchaya, M. J. Claesson, F. He, M. Mantzourani, L. Mulas, A. Ferrarini, B. Gao, M. Delledonne, B. Henrissat, P. Coutinho, M. Oggioni, R. S. Gupta, Z. Zhang, D. Beighton, G. F. Fitzgerald, P. W. O'Toole, and D. van Sinderen. The *Bifidobacterium dentium Bd1* genome sequence reflects its genetic adaptation to the human oral cavity. *PLoS Genet*, 5:e1000785, 2009.

[71] K. M. Wassarman and G. Storz. 6S RNA regulates *E. coli* RNA polymerase activity. *Cell*, 101:613–623, 2000.

[72] R. H. Waterston, K. Lindblad-Toh, E. Birney, J. Rogers, J. F. Abril, P. Agarwal, R. Agarwala, R. Ainscough, M. Alexandersson, P. An, S. E. Antonarakis, J. Attwood, R. Baertsch, J. Bailey, K. Barlow, S. Beck, E. Berry, B. Birren, T. Bloom, P. Bork, M. Botcherby, N. Bray, M. R. Brent, D. G. Brown, S. D. Brown, C. Bult, J. Burton, J. Butler, R. D. Campbell, P. Carninci, S. Cawley, F. Chiaromonte, A. T. Chinwalla, D. M. Church, M. Clamp, C. Clee, F. S. Collins, L. L. Cook, R. R. Copley, A. Coulson, O. Couronne, J. Cuff, V. Curwen, T. Cutts, M. Daly, R. David, J. Davies, K. D. Delehaunty, J. Deri, E. T. Dermitzakis, C. Dewey, N. J. Dickens, M. Diekhans, S. Dodge, I. Dubchak, D. M. Dunn, S. R. Eddy, L. Elnitski, R. D. Emes, P. Eswara, E. Eyras, A. Felsenfeld, G. A. Fewell, P. Flicek, K. Foley, W. N. Frankel, L. A. Fulton, R. S. Fulton, T. S. Furey, D. Gage, R. A. Gibbs, G. Glusman, S. Gnerre, N. Goldman, L. Goodstadt, D. Grafham, T. A. Graves, E. D. Green, S. Gregory, R. Guig, M. Guyer, R. C. Hardison, D. Haussler, Y. Hayashizaki, L. W. Hillier, A. Hinrichs, W. Hlavina, T. Holzer, F. Hsu, A. Hua, T. Hubbard, A. Hunt, I. Jackson, D. B. Jaffe, L. S. Johnson, M. Jones, T. A. Jones, A. Joy, M. Kamal, E. K. Karlsson, D. Karolchik, A. Kasprzyk, J. Kawai, E. Keibler, C. Kells, W. J. Kent, A. Kirby, D. L. Kolbe, I. Korf, R. S. Kucherlapati, E. J. Kulbokas, D. Kulp, T. Landers, J. P. Leger, S. Leonard, I. Letunic, R. Levine, J. Li, M. Li, C. Lloyd, S. Lucas, B. Ma, D. R. Maglott, E. R. Mardis, L. Matthews, E. Mauceli, J. H. Mayer, M. McCarthy, W. R. McCombie, S. McLaren, K. McLay, J. D. McPherson, J. Meldrim, B. Meredith, J. P. Mesirov, W. Miller, T. L. Miner, E. Mongin, K. T. Montgomery, M. Morgan, R. Mott, J. C. Mullikin, D. M. Muzny, W. E. Nash, J. O. Nelson, M. N. Nhan, R. Nicol, Z. Ning, C. Nusbaum, M. J. O'Connor, Y. Okazaki, K. Oliver, E. Overton-Larty, L. Pachter, G. Parra, K. H. Pepin, J. Peterson, P. Pevzner, R. Plumb, C. S. Pohl, A. Poliakov, T. C. Ponce, C. P. Ponting, S. Potter, M. Quail, A. Reymond, B. A. Roe, K. M. Roskin, E. M. Rubin, A. G. Rust, R. Santos, V. Sapojnikov, B. Schultz, J. Schultz, M. S. Schwartz, S. Schwartz, C. Scott, S. Seaman, S. Searle, T. Sharpe, A. Sheridan, R. Shownkeen, S. Sims, J. B. Singer, G. Slater, A. Smit, D. R. Smith, B. Spencer, A. Stabenau, N. Stange-Thomann, C. Sugnet, M. Suyama, G. Tesler, J. Thompson, D. Torrents, E. Trevaskis, J. Tromp, C. Ucla, A. Ureta-Vidal, J. P. Vinson, A. C. Von Niederhausern, C. M. Wade, M. Wall, R. J. Weber, R. B. Weiss, M. C. Wendl, A. P. West, K. Wetterstrand, R. Wheeler, S. Whelan, J. Wierzbowski, D. Willey, S. Williams, R. K. Wilson, E. Winter, K. C. Worley, D. Wyman, S. Yang, S. P. Yang, E. M. Zdobnov, M. C. Zody, and E. S.

Lander. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420:520–562, 2002.

[73] Z. Weinberg, J. E. Barrick, Z. Yao, A. Roth, J. N. Kim, J. Gore, J. X. Wang, E. R. Lee, K. F. Block, N. Sudarsan, S. Neph, M. Tompa, W. L. Ruzzo, and R. R. Breaker. Identification of 22 candidate structured RNAs in bacteria using the CMfinder comparative genomics pipeline. 35:4809–4819, 2007.

[74] Z. Weinberg, J. Perreault, M. M. Meyer, and R. R. Breaker. Exceptional structured noncoding RNAs revealed by bacterial metagenome analysis. *Nature*, 462:656–659, 2009.

[75] Z. Weinberg, J. X. Wang, J. Bogue, J. Yang, K. Corbino, R. H. Moy, and R. R. Breaker. Comparative genomics reveals 104 candidate structured RNAs from bacteria, archaea, and their metagenomes. *Genome Biol*, 11:R31, 2010.

[76] D. Yusuf, M. Marz, P. F. Stadler, and I. L. Hofacker. Bcheck: a wrapper tool for detecting RNase P RNA genes. *BMC Genomics*, 11:432, 2010.

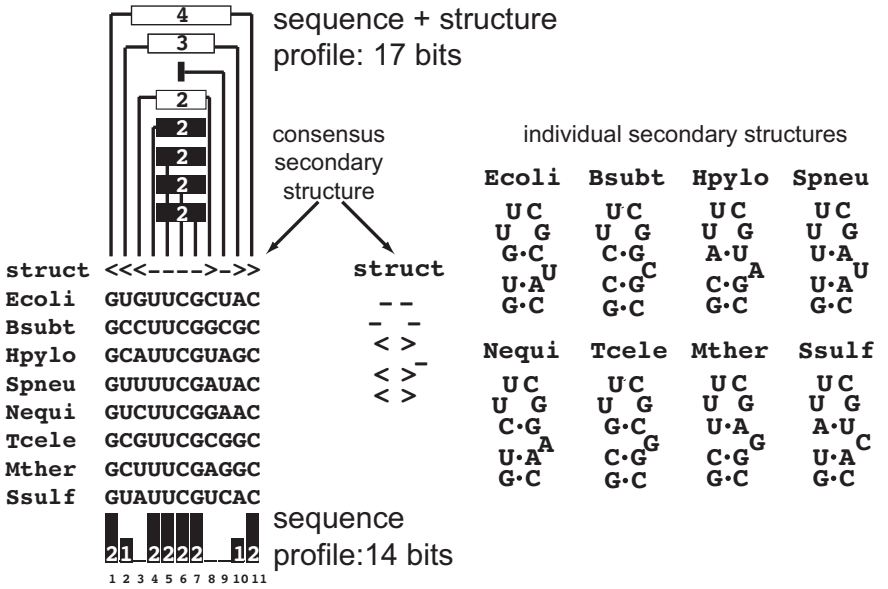


Fig. 1 Information in a sequence-only versus a sequence and structure profile. The eight sequence alignment for a fabricated RNA family used to build both of the profiles is on the left. The `struct` line denotes the consensus secondary structure of the family, with basepaired columns indicated by matching nested < and > characters and connected by lines at top of figure. The structure is ignored by the sequence-only profile but used in the sequence and structure profile to define dependencies between basepaired columns. The eight individual secondary structures, defined by imposing the consensus structure on each sequence, are shown on the right. Boxes with internal numbers at top and bottom of the alignment indicate the number of bits per position from the sequence (black), or per basepair from the structure (white). This figure is similar to one from [19].

Table 1 Summary of RNA annotations in published genomes. Counts were taken from "NCBI Genome" RefSeq annotation for the listed genomes (`www.ncbi.nlm.nih.gov/sites/genome`). Archaeal and bacterial genomes were selected as the first five published in 2010 according to NCBI (`www.ncbi.nlm.nih.gov/genomes/lproks.cgi`) for which a Refseq entry and a referenced publication was available, as of March 30, 2011. Eukaryotes were selected from to be representative, from "complete" genomes according to NCBI (`www.ncbi.nlm.nih.gov/genomes/leuks.cgi`) as of April 29, 2011. One genome from each "group" (fungi, protists, plant, animal) was chosen. "rRNA" includes 5S, SSU, LSU, and 5.8S for eukaryotes only. Abbreviations: TIGR: The Institute for Genomic Research, RAS: Russian Academy of Sciences, JGI: Joint Genome Institute, NML: National Microbiology Laboratory. GenBank accessions for archaeal and bacterial genomes: *M. rum.*: CP001719.1, *H. vol.*: CP001956.1, *H. jeo.*: CP002062.1, *A. sac.*: CP001742.1, *M. mar.*: CP001710.1, *C. rod.*: FN543502.1, *B. den.*: CP001750.1, *P. sta.*: CP001848.1, *L. mon.*: CP002062.1, *C. ucy.*: CP001602.1. NCBI Genome project RefSeq ID for eukaryotic genomes: *C. dub.*: 38659, *L. bra.*: 19185, *A. tha.*: 116, *M. mus.*: 169.

organism [reference]	sequencing center, country	tRNAs method	#	rRNAs method	#	other RNAs method	#
archaea							
Methanobrevibacter ruminantium [46]	AgResearch, New Zealand	tRNAscan-SE	58	BLASTN	8		0[a]
Haloferax volcanii [34]	TIGR, USA	tRNAscan-SE	52	BLASTN	6		0[a]
Halalkalicoccus jeotgali [64]	Kyung Hee Univ., Korea	tRNAscan-SE	49	RNAmmer	3		0
Acidilobus saccharovorans [54]	RAS, Russia	tRNAscan-SE	45	RNAmmer	3		0
Methanothermobacter marburgensis [49]	G. August Univ. Germany	tRNAscan-SE	40	RNAmmer	2	*unknown*	2
bacteria							
Citrobacter rodentium [62]	Sanger Institute, UK	tRNAscan-SE	86	*unknown*	22	Infernal & Rfam	56
Bifidobacterium dentium [70]	Univ. of Parma, Italy	tRNAscan-SE	55	BLASTN	13		0
Pirellula staleyi [13]	JGI, USA	*unknown*	46	*unknown*	3	*unknown*	3
Listeria monocytogenes [30]	NML, Canada	tRNAscan-SE	58	RNAmmer	15		0
Cyanobacterium UCYN-A [69]	UC Santa Cruz, USA	tRNAscan-SE	36	search_for-_rnas	6		0
eukaryotes							
Candida dubliniensis [39]	Sanger Institute UK	*unknown*	101		0	*unknown*	11
Leishmania braziliensis [60]	Sanger Institute UK		0		0	*unknown*	6
Arabidopsis thaliana [68][b]	multiple centers	tRNAscan-SE, tRNAscan	688	BLASTN	14	*unknown*	689
Mus musculus [72]	multiple centers	tRNAscan-SE	509[a]	*unknown*	5	*unknown*	4059[a]

[a] Numbers from NCBI (shown here) are inconsistent with explicitly mentioned counts given in the referenced publication for this genome.

Table 2 Some popular family-specific tools for identifying RNAs in genomes.

program	type of RNA	prefilter stage	final stage	reference
tRNAscan-SE	tRNA	sequence-based; tRNA-specific	CMs	[51]
Aragorn	tRNA, tmRNA	*none*	tRNA/tmRNA-specific heuristic	[45, 43]
Arwen	mitochondrial tRNA	*none*	mito tRNA-specific heuristic	[44]
RNAmmer	rRNA (5S,5.8S,SSU,LSU)	small "spotter" profile HMMs	profile HMMs (full-length)	[42]
SRPscan	SRP RNA	sequence/structure pattern (RNABOB)	CMs	[63]
Bcheck	RNase P RNA	sequence/structure pattern (RNABOB)	CMs	[76]

Table 3 Infernal predicted RNAs in the archaeon *Methanobrevibacter ruminantium* (GenBank accession CP001719.1). All Rfam 10.1 families [28] for which Infernal finds at least one hit above the Rfam bit-score gathering threshold (GA) are shown. Also shown is LSU rRNA, which is not in Rfam as discussed in the text. Non-obvious column heading descriptions: "# in GenBank": number of RNAs in GenBank annotation; "believed": hits believed to be real homologs, these are all nonoverlapping hits with E-values below 0.01 except for the single CRISPR-DR42 hit ($E = 0.0045$) which is likely a false positive (the choice of 0.01 here is discussed in section 2.1); "unique": number of nonoverlapping Infernal hits, overlaps of more than 50% the length of the shorter sequence were removed by keeping the hit with the lowest E-value amongst the overlapping hits; "total": total Infernal hits, including overlaps; "best hit": bit scores and E-values of the best scoring hits out of total hits for each family. The following sets of families shared overlapping hits, the family with the lowest E-value for all overlaps is listed first: SSU_rRNA_archaea and SSU_rRNA_bacteria (2 hits), CRISPR-DR2 and CRISPR-DR39 (60 hits), and sR2, sR1, and snoPyro_CD (1 hit).

Rfam family ID	Rfam type	Rfam GA bit thresh	# in Gen-bank	Infernal hits above Rfam GA thr				
				# hits			best hit	
				believed	unique	total	bit	E-value
tRNA	Gene; tRNA;	24.0	58	59	59	59	69.8	7.5e-16
5S_rRNA	Gene; rRNA;	16.0	4	3	3	3	58.9	1.4e-11
LSU rRNA			2					
SSU_rRNA_archaea	Gene; rRNA;	658.0	2	2	2	2	1483.0	0
SSU_rRNA_bacteria	Gene; rRNA;	600.0	0	0	0	2	1090.7	0
Archaea_SRP	Gene;	87.0	0	1	1	1	183.7	6.2e-52
RNaseP_arch	Gene; ribozyme;	53.0	0	1	1	1	193.9	3.5e-63
FMN	Cis-reg; riboswitch;	40.0	0	1	1	1	110.8	8.6e-28
CRISPR-DR2	Gene; CRISPR;	22.0	0	60	61	61	28.2	0.0043
CRISPR-DR39	Gene; CRISPR;	20.0	0	0	2	62	26.5	0.019
CRISPR-DR42	Gene; CRISPR;	19.2	0	0	1	1	20.2	0.0045
sR2	Gene; snRNA; snoRNA; CD-box;	20.0	0	1	1	1	27.5	9e-06
sR1	Gene; snRNA; snoRNA; CD-box;	21.0	0	0	0	1	23.7	0.00017
snoPyro_CD	Gene; snRNA; snoRNA; CD-box;	20.0	0	0	0	1	27.1	0.0018
sR11	Gene; snRNA; snoRNA; CD-box;	16.0	0	0	1	1	16.8	0.021
total			66	128	136	200		

Table 4 Comparison of predictions by Infernal and family-specific methods for various RNAs in five archaeal genomes. Abbreviated genome names are fully listed in Table 1. Genome sizes in millions of bases (Mb) are shown in parantheses underneath genome names. Columns labeled "hits" include total number of predictions, and those labeled "unq" include unique hits that are not found with the method on the adjacent line. Average timings are reported in seconds ("(secs)"). The following Rfam 10.1 models were used for each Infernal search: "tRNA": RF00005, "RNase P RNA": RF00373, "SRP RNA": RF01857, "5S rRNA": RF00001, "SSU rRNA": RF01959. LSU rRNA Infernal searches were not performed because Rfam 10.1 has no LSU model. All programs were run in default mode, except when options were necessary to restrict searches to the specific family and/or domain being tested. SRPscan was run in fast mode with non-Alu models. Infernal's cmsearch was run with the `--ga` option which sets the reporting bit score threshold as the family-specific Rfam GA cutoff discussed in the text. Program versions used: Infernal v1.1rc1; tRNAscan-SE v1.23; Aragorn v1.2; Bcheck v0.6; web version of SRPscan available at `http://bio.lundberg.gu.se/srpscan/`; RNAmmer v1.2. Because no downloadable version of SRPscan was available, times were measured manually via stopwatch on their website and so are approximate. All other times were measured as single execution threads on 2.66 GHz Intel Xeon Gainestown (X5550) processors.

		organism (archaea)										
		M. rum. (2.9 Mb)		*H. vol.* (4.0 Mb)		*H. jeo.* (3.7 Mb)		*A. sac.* (1.5 Mb)		*M. mar.* (1.6 Mb)		avg time
family	software	hits	unq	hits	unq	hits	unq	hits	unq	hits	unq	(secs)
tRNA	Infernal	59	2	49	1	46	0	43	0	38	0	2.2
	tRNAscan-SE	58	1	51	3	49	3	45	2	40	2	27.6
tRNA	Infernal	59	3	49	2	46	3	43	5	38	1	2.2
	Aragorn	56	0	54	7	48	5	39	1	38	1	0.9
RNase P RNA	Infernal	1	1	1	0	1	0	1	0	1	0	28.8
	Bcheck	0	0	1	0	1	0	1	0	1	0	13.7
SRP RNA	Infernal	1	1	1	0	1	0	1	0	1	0	10.0
	SRPscan	0	0	1	0	1	0	1	0	1	0	8.0
5S rRNA	Infernal	3	3	2	0	1	0	1	1	3	3	2.0
	RNAmmer	0	0	2	0	1	0	0	0	0	0	16.1
SSU rRNA	Infernal	2	0	2	0	1	0	1	0	2	0	31.7
	RNAmmer	2	0	2	0	1	0	1	0	2	0	16.4
LSU rRNA	Infernal											
	RNAmmer	2	2	2	2	1	1	1	1	2	2	18.8

Table 5 Comparison of predictions by Infernal and family-specific methods for various RNAs in five bacterial genomes. Abbreviated genome names are fully listed in Table 1. Column headings and program versions, options and cutoffs are the same as described in caption of Table 4, except that for SRPscan, "rare TRRC tetraloop" was used for *P. staleyi* and "common GRRA tetraloop" was used for all others. Rfam 10.1 models used for each Infernal search: "tRNA": RF00005, "tmRNA": RF00023, "RNase P RNA": RF00010 and RF00011, "SRP RNA": RF00169 and RF01854, "5S rRNA": RF00001, "SSU rRNA": RF00177.

		organism (bacteria)										
		C. rod. (5.4 Mb)		*B. den.* (2.6 Mb)		*P. sta.* (6.2 Mb)		*L. mon.* (3.1 Mb)		*C. ucy.* (1.4 Mb)		avg time
family	software	hits	unq	hits	unq	hits	unq	hits	unq	hits	unq	(sec)
tRNA	Infernal	85	1	57	1	46	2	57	0	37	0	2.0
	tRNAscan-SE	84	0	56	0	46	2	58	1	37	0	34.5
tRNA	Infernal	85	1	57	1	46	1	57	0	37	0	2.0
	Aragorn	87	3	56	0	49	4	59	2	37	0	1.1
tmRNA	Infernal	1	0	1	0	1	0	2	1	1	0	40.9
	Aragorn	1	0	1	0	1	0	1	0	1	0	2.3
RNase P RNA	Infernal	1	0	1	0	1	0	2	0	1	0	28.0
	Bcheck	1	0	1	0	1	0	1	0	1	0	12.8
SRP RNA	Infernal	1	0	1	0	1	0	2	0	1	0	29.2
	SRPscan	1	0	1	0	1	0	1	0	1	0	4.0
5S rRNA	Infernal	8	0	6	1	1	0	5	0	2	0	2.9
	RNAmmer	8	0	5	0	1	0	5	0	2	0	20.2
SSU rRNA	Infernal	7	0	4	0	1	0	5	0	2	0	34.0
	RNAmmer	7	0	4	0	1	0	5	0	2	0	20.4
LSU rRNA	Infernal											
	RNAmmer	7	7	4	4	1	1	5	5	2	2	27.9

Table 6 Comparison of predictions by Infernal and family-specific methods for various RNAs in four eukaryotic genomes. Abbreviated genome names are fully listed in Table 1. RNAmmer does not do 5.8S rRNA searches, so the corresponding cells are left blank in the table. Column headings and program versions, options and cutoffs are the same as described in caption of Table 4, except that for Infernal searches only hits with bit scores above the Rfam GA cutoff *and* E-values below $1e-5$ were considered. Rfam 10.1 models used for each Infernal search: "tRNA": RF00005, "RNase P RNA": RF00009, "5S rRNA": RF00001, "SSU rRNA": RF01960,"5.8S rRNA": RF00002.

		organism (eukarya)								
		C. dub. (14.6 Mb)		*L. bra.* (31.4 Mb)		*A. tha.* (119.7 Mb)		*M. mus.* (2654.9 Mb)		avg time
family	software	hits	unq	hits	unq	hits	unq	hits	unq	(secs)
tRNA	Infernal	123	0	82	0	676	5	442	11	359.8
	tRNAscan-SE	130	7	82	0	699	28	26201[a]	25770	3481.5
tRNA	Infernal	123	36	82	1	676	44	442	14	359.8
	Aragorn	89	2	85	4	666	34	1656	1228	130.0
RNase P RNA	Infernal	1	1	0	0	0	0	17	11	651.1
	Bcheck	0	0	0	0	0	0	6	0	218.4
5S rRNA	Infernal	2	0	9	1	497	3	115	115	134.3
	RNAmmer	2	0	8	0	498	4	0	0	3741.2
SSU rRNA	Infernal	1	0	0	0	4	0	2	0	3203.0
	RNAmmer	1	0	0	0	5	1	2	0	3450.2
LSU rRNA	Infernal									
	RNAmmer	1	1	0	0	4	4	3	3	3816.3
5.8S rRNA	Infernal	1	1	0	0	2	2	2	2	173.3
	RNAmmer									

[a] 22918 of these are annotated as pseudogenes by tRNAscan-SE.

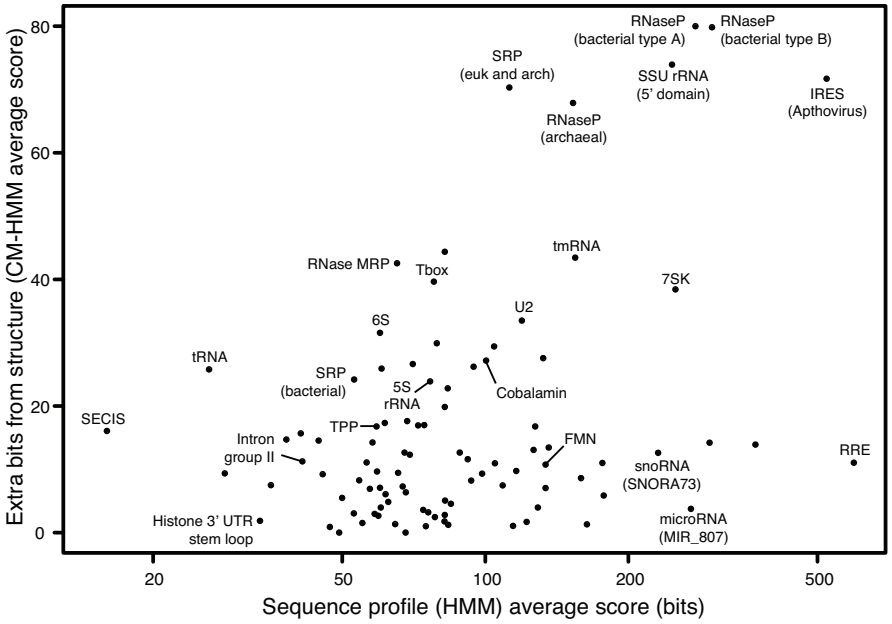


Fig. 2 Additional information (in bits) gained by sequence and structure profiles (CMs) versus sequence-only profiles (HMMs) for various RNA families. Sequence and structure profiles are most advantageous for families with less primary sequence information (towards left) and more secondary structure information (towards top), so Rfam families that gain the most from including secondary structure terms in a homology search are those toward the upper left quadrant. Data shown for the 95 Rfam release 9.1 [29] families with 50 or more sequences in the *seed* alignment. For each family, the seed alignment was used to build two profile models, a CM and a profile HMM. From each model, 10,000 sequences were generated and scored, and the average score per sampled sequence was calculated. Several of the outlying points are labeled by the name of RNA family as given by Rfam. Note that the x-axis is drawn on a log scale. Models were built and sequences were generated and scored using Infernal version 1.0 programs cmbuild, cmemit and cmalign.

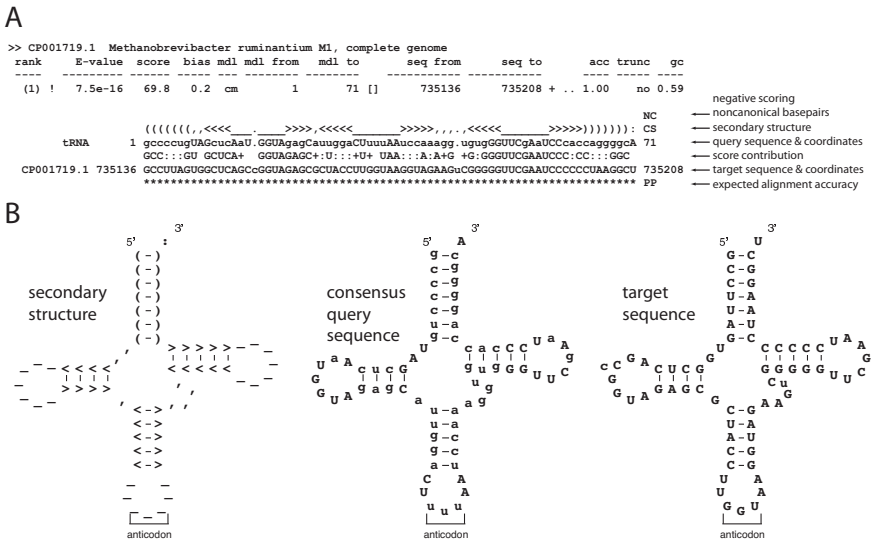
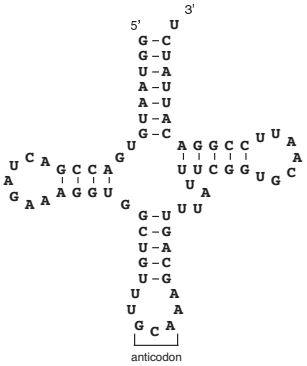


Fig. 3 A sequence- and structure-based alignment of a predicted tRNA in *M. ruminantium* to the Rfam 10.1 tRNA CM. (A) Raw output from cmsearch showing the alignment of positions 735136 to 735208 of the target sequence CP001719.1 (renamed from gi|288541968|gb|CP001719.1| to save space) to the model. Scores and alignment annotation are explained in the text. (B) Secondary structure diagrams of the consensus tRNA structure annotation from (A), the consensus query tRNA sequence from the CM, and the predicted tRNA homolog from the target genome.

```
>> CP001719.1  Methanobrevibacter ruminantium M1, complete genome
 rank     E-value  score  bias mdl mdl from   mdl to      seq from      seq to       acc trunc   gc
 ----   --------- ------ ----- --- -------- --------  ---------- ----------       ----- ----- ----
 (54) !     1e-08   44.7   0.2  cm        1       71 []      361145      361075 - .. 0.99    no 0.44

                                                                                                NC
                   (((((((,,<<<<_______>>>>,<<<<<_______>>>>>,,,,<<<<<_______>>>>>)))))))): CS
         tRNA      1 gccccugUAGcucAaUGGUAgagCauuggaCUuuuAAuccaaaggugugGGUUCgAaUCCcaccaggggcA 71
                     G :::UGU :C::A U G A::G:+ ::G +U+ +AA C:: + +U: :GGU C+A+UCC: :CA::: C
  CP001719.1 361145 GGUAAUGUGACCGACUAGAAAGGUGGCUGUUUGCAAAGCAGUUUAUUUCGGUGCAAUUCCGGACAUUAUCU 361075
                     ***************888************************************************** PP
```




```
>> CP001719.1  Methanobrevibacter ruminantium M1, complete genome
 rank     E-value  score  bias mdl mdl from   mdl to      seq from      seq to       acc trunc   gc
 ----   --------- ------ ----- --- -------- --------  ---------- ----------       ----- ----- ----
 (57) !   4.7e-08   42.3   0.1  cm        1       71 []      363803      363716 - .. 0.98    no 0.50

                                                                                                                NC
                   (((((((,,..<<<<___..___..._>>>>,<<<<<_______>>>>>,,.............,.,<<<<<_______>>>>>)))))))): CS
         tRNA      1 gccccugUA..GcucAaU..GGU..AgagCauuggaCUuuuAAuccaaag............g.ugugGGUUCgAaUCCcaccaggggcA 71
                     ::C:::GUA  GC::A+U  GGU  A::GCA :G:ACUU+  AU:C:                 +:UGGGUUC+AAUCCCA:C:::G::A
  CP001719.1 363803 AGCUGUGUAguGCCAAGUcaGGUcuAAGGCAGCGGACUUAGGAUUCGUG-gaguuagucuca-uCAUGGGUUCAAAUCCCAUCGCAGCUA 363716
                     ********9889*************************************.8**********5.5************************* PP
```

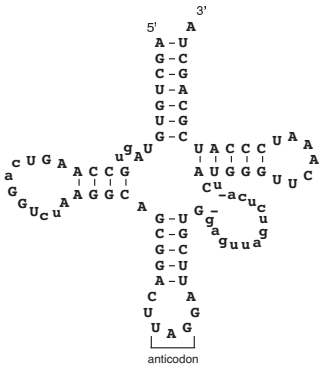


Fig. 4 Two Infernal-predicted tRNAs in the *Methanobrevibacter ruminantium* genome that are not predicted by tRNAscan-SE. The target sequence CP001719.1 has been renamed from gi|288541968|gb|CP001719.1| to save space. The cmsearch output for each alignment to the Rfam 10.1 tRNA model is shown above the corresponding predicted secondary structure. Nucleotides inserted relative to the Rfam consensus model are in lowercase.