RNA Secondary Structure Analysis Using the Vienna RNA Package

The Vienna RNA package (Hofacker et al., 1994) is a free software package that implements a variety of algorithms for the prediction and analysis of RNA secondary structures. The various algorithms are usually accessed through several command-line programs (discussed here), but the package also provides a C library that can be used to develop new programs, as well as a Perl module that gives access to all functions of the library from the Perl scripting language.

For structure prediction (see Basic Protocol 1), the package implements the classic minimum free energy algorithm of Zuker and Stiegler (1981), the partition function algorithm of McCaskill (1990), which calculates base pair probabilities in thermodynamic equilibrium, and the suboptimal folding algorithm (Wuchty et al., 1999), which generates all suboptimal structures within a given energy range of the optimal energy.

If several sequences are expected to share a common structure, highly accurate predictions of the consensus structure can be obtained by combining thermodynamic rules with an analysis of sequence variation and covariation. Such a method is implemented in the RNAalifold program (Hofacker et al., 2002; see Basic Protocol 2).

Finally, the authors of the Vienna RNA package provide an algorithm for inverse folding, i.e., to design sequences with a predefined structure (see Basic Protocol 3).

NOTE: Investigators who are unfamiliar with the Unix environment should refer to *APPENDIX 1C* and *APPENDIX 1D*.

USING THE RNAfold PROGRAM TO PREDICT RNA SECONDARY STRUCTURE

BASIC PROTOCOL 1

Secondary structure prediction from individual sequences is the most frequently performed task. Basic structure prediction is done using the RNAfold program; for short sequences the RNAsubopt program can also be used. The programs support quite a few options that modify the way the prediction is done. Here, only the default settings will be used; all other options are described in detail on the RNAfold main page, and a few are further discussed in the Commentary of this unit (see Critical Parameters and Troubleshooting).

Necessary Resources

Hardware

A personal computer running Linux is recommended; a Unix workstation (e.g., from Sun, SGI, or IBM) or Macintosh under OS X may be used, but these platforms are less well tested. PCs with MS Windows require significant extra installation effort. For predictions on long sequences, sufficient memory should be available: e.g., a complete HIV genome will require ~1 Gb of memory.

Software

Vienna RNA package (see Support Protocol)

A basic *x-y* plotting program (e.g., xmgrace; *http://plasma-gate.weizmann.ac.il/Grace/*) for mountain plots; an alternative for use on most Unix systems would be gnuplot (*http://www.gnuplot.info*)

Contributed by Ivo L. Hofacker

Current Protocols in Bioinformatics (2003) 12.2.1-12.2.12

Files

One or more RNA sequences. The RNAfold program uses a "trivial" sequence format with each sequence on a single line without embedded whitespace. Each sequence may be preceded by a line starting with the > character followed by a sequence name, which will be used for output filenames later. Thus, sequences in FASTA format (*APPENDIX 1B*) can be converted simply by removing whitespace and newlines within the sequence. For sequence files in other formats, the program Readseq (*APPENDIX 1E*) can be used. A modified version of Readseq that writes output suitable for RNAfold is included in the package. Lowercase characters will be converted to uppercase and T's will be replaced by U's. Any remaining characters except for A, C, G, U, I, X, and K will be treated as nonpairing bases (*APPENDIX 1A*).

1. Download and install the Vienna RNA package (see Support Protocol).

Prepare the sequence file for input

2a. *To compute a single optimal secondary structure (i.e., a structure with minimum free energy, mfe):* Assuming that the sequence file of interest is named `file.seq`, type:

```
RNAfold < file.seq > file.fold
```

2b. *To compute optimal (mfe) structure, partition function, and pair probabilities:* Type the command in step 2a and add a `-p` option:

```
RNAfold -p < file.seq > file.fold
```

Note that the program reads from stdin and writes to stdout, i.e., the < and > above are necessary to redirect input and output. It is also possible to start the program without an input file and type the sequence(s) on the terminal, or use the program in a pipe (i.e., have another program produce the input). Depending on the length of the sequences, the computation will take between a fraction of a second (e.g., for tRNA) and several hours (for a complete viral genome).

3. Examine and interpret the output file.

The output file (`file.fold` in our example) first repeats the input sequence; the next line contains the predicted mfe structure in bracket notation and its free energy in kcal/mol (Fig. 12.2.1). In the bracket notation, unpaired positions are represented by dots, while base pairs (i, j) are represented by a pair of matching parentheses at positions i and j. Thus the secondary structure `(((..((((...)))).)))` describes a stem-loop structure consisting of an outer helix of 3 base pairs interrupted by an interior loop of size 3, a second helix of length 4, and a hairpin loop of size 3.

If partition function folding was selected above (step 2b), the next line contains another string giving a condensed representation of the pair probabilities followed by the ensemble free energy in kcal/mol (Fig. 12.2.1). The structure string is similar to the bracket notation but contains additional symbols: parentheses represent positions with strong tendency to pair and dots represent positions that are mostly unpaired, while curly brackets and commas represent positions with less clear pairing preferences. See the manual (http://www.tbi.univie.ac.at/~ivo/RNA/RNAfold.html) for the exact definitions.

From the minimum free energy, E, and the ensemble free energy, F, the frequency of the mfe structure in thermodynamic equilibrium can be computed as:

$$p = \exp\left(\frac{-(E-F)}{RT}\right)$$

This value is given on the last line. The mfe structure is well defined when the difference $E - F$ is small, and the two structure strings look similar. The more well defined the structure, the more confidence one may have in the accuracy of the prediction.

4. View the PostScript figures.

   Apart from the text output, RNAfold produces a PostScript structure drawing, suitable for inclusion in publications as well as for printing on any PostScript-capable printer (Fig. 12.2.1). For on-screen, viewing a PostScript viewer such as GhostScript (or one of its front ends, i.e., gv or gsview; http://www.cs.wisc.edu/~ghost/) is needed. If the input defined a sequence name (say `seq1`*), it will be used to name the PostScript file (e.g,.* `seq1 ss.ps`*); otherwise the default filename* `rna.ps` *will be used.*

   Pair probabilities will be written in the form of a PostScript "dot plot." The dot plot shows a $n \times n$ matrix of squares, such that the area of the square at row i and column j in the upper right half is proportional to probability of the pair (i, j), while the lower left half shows all pairs belonging to the mfe structure. The name of the dot plot file will again be derived from the sequence name (e.g., `seq1 dp.ps`*) or the default filename* `dot.ps` *will be used.*

   Dot plots are an excellent way to visualize structural alternatives. For an RNA with well-defined mfe structure, the upper right half should only contain a few small additional dots compared to the lower left. The PostScript dot plot is constructed such that the actual pair probabilities can be easily read from the file itself (see, e.g., step 5).

5. Produce a mountain plot.

   Secondary structure graphs and dot plots both become cumbersome for long file sequences. A mountain plot is a structure representation that works well even for long sequences, and which is well suited for comparing structures. A mountain plot is an x-y graph that plots the number of base pairs enclosing a sequence position, or, for pair probabilities, the average number of enclosing pairs. The Perl script `mountain.pl` *can be used to produce the coordinates for a mountain plot from a dot plot PostScript file. The result can then be plotted with any x-y plotting program. Using, e.g., the xmgrace plotting program, the following command is typed:*

```
mountain.pl seq1_dp.ps | xmgrace -pipe
```

   If a `mountain.pl: Command not found` *error is encountered, use the full path in the command (e.g.,* `/usr/local/share/ViennaRNA/bin/mountain.pl`*).*

   The resulting plot shows three curves: two mountain plots derived from mfe structure and pair probabilities and a positional entropy derived from the pair probabilities:

$$S_i = -\sum_j p_{ij} \log p_{ij} - p_i^u \log p_i^u$$

   where p_i^u is the probability of i being unpaired. Well-defined regions are marked by low entropy.

6. Include experimental constraints.

   Secondary structure prediction is of course error-prone, and no prediction should be trusted blindly without experimental support. If any experimental results (such as chemical probing data) are available, it is possible to test whether the prediction is compatible with the experimental data. Furthermore, constraints can be used to ensure that RNAfold will only consider structures compatible with the constraints.

   To do constrained folding, open the sequence file in a text editor and add another line after the sequence consisting of the symbols `x`*,* `|`*,* `.`*, and matching parentheses,* `()`*. A pair of matching parentheses signify that the corresponding positions must form a base pair. A vertical line (* `|` *) marks a position that must pair, and an* `x` *marks a position that must not pair. The dot (* `.` *) marks positions without constraint. Refold the sequences with constraints using the* `-C` *option:*

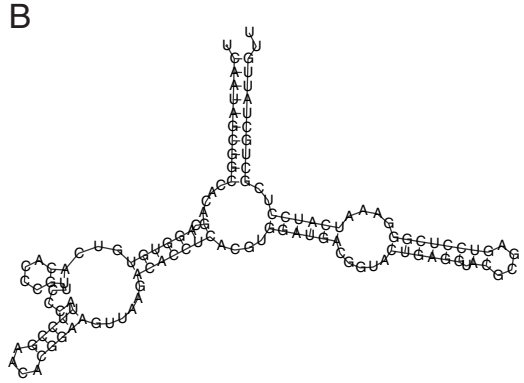
```
RNAfold -p -C < file_c.seq > file_c.fold
```

   One can now compare the constrained and unconstrained foldings. Ideally, the constraints should only lead to a small change in energy.

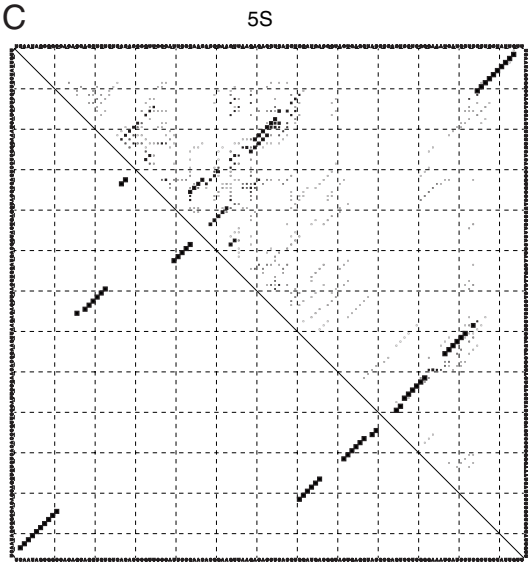
A

```
% cat 5s.seq
> 5s
UCAAUAGCGGCCACAGCAGGUGUGUCACACCCGUUCCCAUUCCGAACACGGAAGUUAAGACACCUCACGUGGAUGACGGUACUGAGGUACGCGAGUCCU
% RNAfold -p < 5s.seq > 5s.fold
RNAfold -p < 5s.seq > 5s.fold
moritz ~/TeX/CP/fig> cat 5s.fold
% 5s
UCAAUAGCGGCCACAGCAGGUGUGUCACACCCGUUCCCAUUCCGAACACGGAAGUUAAGACACCUCACGUGGAUGACGGUACUGAGGUACGCGAGUCCU
.((((((((((....(.(((((((...((....)).....(((((....))))).....)))))))....(((((.....(((((((.((....)))))))))....)))))))..)))))))).. (-47.70)
.((((((((((....{..({{{(.[.......({{{.......|}...|}}..)}....)))))}}..(.(((((.....(((((((.((....)))))))))....)))))))..)))))))).. [-52.08]
 frequency of mfe structure in ensemble 0.000814848       0.000814848
% gv 5s_ss.ps
% gv 5s_dp.ps
% mountain.pl 5s_dp.ps | xmgrace -p1pe
```

B



C



D

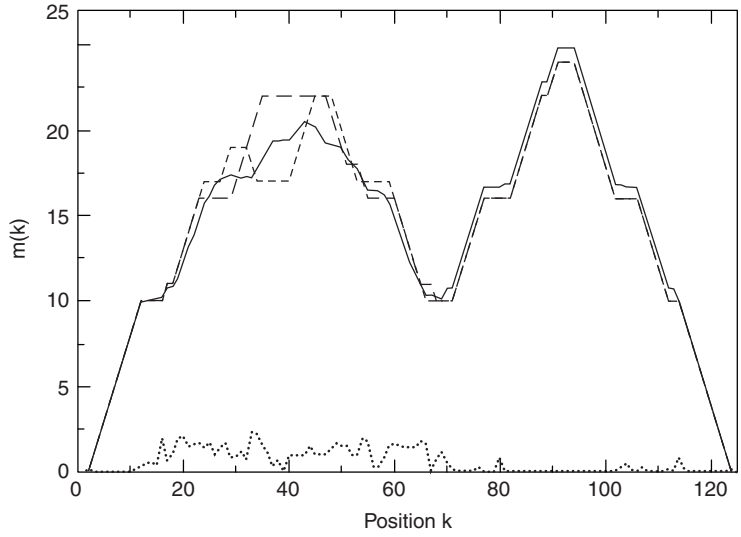


Figure 12.2.1 Sample session, computing the structure of an 5S rRNA. (**A**) Input; (**B**) secondary structure graph, (**C**) dot plot, and (**D**) mountain plot. Colors have been converted to patterns in this black and white reproduction of the mountain plot: solid line (black) represents pair probabilities; short dashes (red) represent mfe structure; dotted line (green) represents positional entropy; and long dashes (blue) represent the correct structure (for comparison).

7. Generate structures with suboptimal folding.

   For short sequences the RNAsubopt program can be used to produce all secondary structures within given energy increment of the mfe. Note that this is quite different from the suboptimal folding offered Micahel Zuker's mfold program (Zuker, 1989). For example the command line:

```
RNAsubopt -e 3 -s < file.seq > file.sub
```

   will generate all secondary structures with energies within 3 kcal/mol of the mfe as an energy sorted list (`-s`*). Since the number of such suboptimal structures grows exponentially with sequence length, this approach is useful only for short sequences (say < 100 nt) and small energy intervals.*

   The `-noLP` *option will cause RNAsubopt to only produce structures without isolated base pairs. This is very useful to keep the number of suboptimal structures manageable.*

COMPUTING CONSENSUS STRUCTURES

BASIC PROTOCOL 2

Functional RNA molecules often exhibit secondary structures that are much better conserved than their sequences. This makes it possible to infer the conserved structure from sequence covariation. RNAalifold is a new program in version 1.5 of the Vienna RNA package. It uses modified dynamic programming algorithms that combine the standard energy model with a covariance term (Hofacker et al., 2002). The accuracy of the predicted consensus structures is much higher than for predictions from single sequences.

The program is used much the same way as RNAfold (see Basic Protocol 1), except that it uses a sequence alignment instead of a single sequence as input.

Necessary Resources

Hardware

A personal computer running Linux is recommended; a Unix workstation (e.g., from Sun, SGI, or IBM) or Macintosh under OS X may be used, but these platforms are less well tested. PCs with MS Windows require significant extra installation effort.

Software

Vienna RNA package (see Support Protocol)
Optional: Perl Tk extension for using the dot plot viewer

Files

A set of related RNA sequences. RNAalifold uses a multiple sequence alignment in Clustal format as input (*UNIT 2.3*). Note that a good alignment is crucial for the quality of the predicted consensus structure. Other alignment programs can be used, as long as they can produce output in Clustal format.

1. Compute the consensus structure from the alignment `file.aln`:

```
RNAalifold -p file.aln > file.alifold
```

2. Examine the output (Fig. 12.2.2).

   The computed structure is written to stdout (here redirected to `file.alifold`*) in the same format used by RNAfold.*

3. Examine the dot plots and structure graphs (Fig. 12.2.2).

   RNAalifold writes three additional output files: the PostScript dot plots and structure graphs `alidot.ps` *and* `alirna.ps`*, and a text file named* `alifold.out`*. The PostScript files look much the same as their equivalents from RNAfold (see Basic Protocol 1), but contain additional information on sequence covariations.*

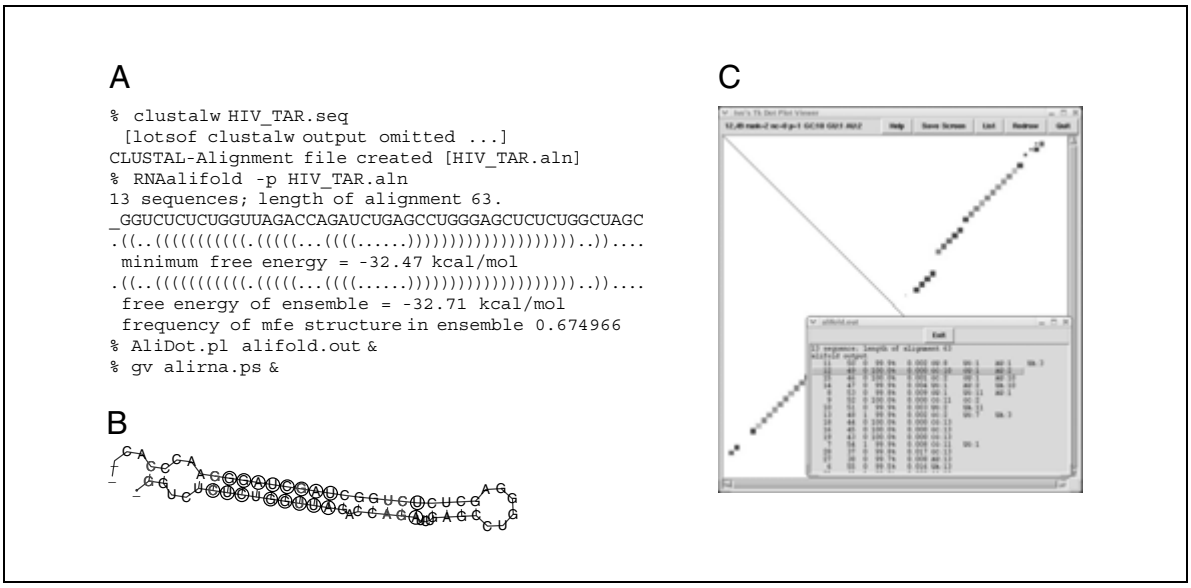


Figure 12.2.2 RNAalifold sample session to predict the consensus structure of the HIV TAR element using the first 60 bases of 13 genomic HIV-1 sequences. (**A**) Command-line input; (**B**) The resulting consensus structure, and (**C**) The AliDot.pl viewer.

In the structure graph `alirna.ps`*, consistent and compensatory mutations are marked by a circle around the variable base(s), i.e., pairs where one pairing partner is encircled exhibit consistent mutations (such as GU → GC), whereas pairs supported by compensatory mutations have both bases marked. Pairs that cannot be formed by some of the sequences are shown in gray instead of black.*

The dot plots produced by RNAalifold use color to convey information on sequence variations. The color hue encodes the number of different base pair types observed, ranging from red for a pair with conserved sequence to blue for a pair where all six pair types (GC, CG, AU, UA, GU, UG) occur. Unsaturated (pale) colors mark pairs that cannot be formed by all sequences.

4. Using the `AliDot.pl` viewer.

   The `alifold.out` *file contains a list with information on all plausible base pairs sorted by the likelihood of the pair. The* `AliDot.pl` *script displays this information in the form of a dot plot equivalent to the PostScript version. The viewer gives feedback and additional information (not available from a PostScript viewer), but requires the Perl Tk module to be installed.*

   Start the viewer using `AliDot.pl alifold.out`*, and a canvas will open showing the dot plot. The + and – keys can be used to zoom in and out. The coordinates of the base pair below the mouse pointer is indicated in the upper left corner. Clicking on any base pair will display more detailed information, including the probability of the pair, the number of sequences unable to form the pair, and the observed base pair types.*

5. Select and refold conserved structure motifs.

   Longer sequences will often exhibit several short conserved structure motifs separated by regions without conserved structure. In this case it is recommended that RNAalifold be rerun on just the conserved regions.

   Identify the conserved region from the dot plot and write a new alignment file for each of them. The ClustalX program (UNIT 2.3) is convenient for cutting a region out of an alignment, but a simple text editor can be used as well.

INVERSE FOLDING

Finding sequences that fold into a predefined structure is the inverse of the structure prediction problem. Often it is useful to design such sequences, e.g., in order to experimentally test an hypothesis about functional structures. The RNAinverse program treats the design as an optimization problem that is tackled using a simple greedy search (i.e., a heuristic search that tries to minimize the distance between the desired structure and the predicted mfe structure).

Necessary Resources

Hardware

A personal computer running Linux is recommended; a Unix workstation (e.g., from Sun, SGI, or IBM) or Macintosh under OS X may be used, but these platforms are less well tested. PCs with MS Windows require significant extra installation effort.

Software

Vienna RNA package (see Support Protocol)

Files

An RNA secondary structure. Input for the RNAinverse program consists of an RNA secondary structure (the target) in bracket notation (on the first line), optionally followed by a sequence to be used as starting point of the optimization (otherwise a random start sequence will be used).

1a. *To run the optimization using only mfe folding:* Type the command:

```
RNAinverse < input
```

The optimization will stop as soon as sequence is found whose mfe structure is the target. This often produces sequences with marginally stable structures. To design several (say, 10) sequences with one call use the `RNAinverse -R 10`.

1b. *To design stable sequences using partition function folding:* Type the command:

```
RNAinverse -Fmp -f 0.1 < input
```

With the option `-Fmp`*, this will first run an mfe optimization as in step 1a, followed by an optimization that tries to maximize the frequency of the target structure in the equilibrium ensemble. The* `-f 0.1` *option will cause the optimization to stop when the difference between the energy of the target and the ensemble free energy is smaller than 0.1 kcal/mol. This corresponds to a frequency of* $p = exp(-0.1/RT) \approx 0.85$.

2. Interpret the output (Fig. 12.2.3).

If successful, output from the mfe part of the calculation will show the designed sequence followed by number of mutations performed by the optimizer. This number is useful as a measure for the ubiquity of the target structure. Since the search is heuristic, there is no guarantee that an exact solution will be found. If the search failed, the output line will end with something like `d=4`*, where the number is a structure distance between the target and the final structure.*

If partition function optimization was selected (`-Fp`*), the next line will display the final sequence followed again by the number of mutations and the frequency of the target structure obtained.*

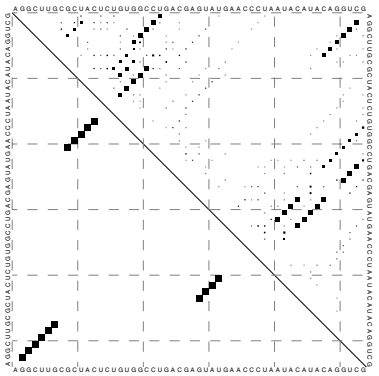
3. Troubleshoot the output.

The ubiquity of secondary structures varies widely. While many valid secondary structure strings never occur as mfe structure of a sequence (i.e., often the design problem has no solution), others are extremely common (see, e.g., Schuster et al., 1994). Correspondingly,

A

```
% cat> hammerhead.struct
.(((((((.(((((...))))).......(((........))))...))))))).
NNNNNNNcNNNNNNNNNNNNNcugaNgaNNNNNNNNNNNNNNNNNNaNNNNNNN
% RNAinverse -Fmp < hammerhead.struct > hammer.inv
% cat hammer.inv
AGGCUUGcGCUACUCUGUGGCcugaCgaGUAUGAACCCUAAUACAUaCAGGUCG   15
CGCCGGCcGCCUCACUGAGGCcugaCgaGCUCGAAAAAAAGAGCAAaGCCGGCA   31  (0.811315)
% RNAalifold -p< hammer.inv
GGCUUGCGCUACUCUGUGGCCUGACGAGUAUGAACCCUAAUACAUACAGGUCG
.(((((((.(((((...))))).......((((.......))))...))))))). (-13.10)
.((((.{.({{{(...)))))}|{....(((({.......})))}}}))))). [-14.42]
 frequency of mfe structure in ensemble 0.190561
CGCCGGCCGCCUCACUGAGGCCUGACGAGCUCGAAAAAAAGAGCAAAGCCGGC
.(((((((.(((((...))))).......(((........))))...))))))). (-26.90)
.(((((((.(((((...))))).......(((........))))...))))))). [-27.33]
 frequency of mfe structure in ensemble 0.811315
```

B



C

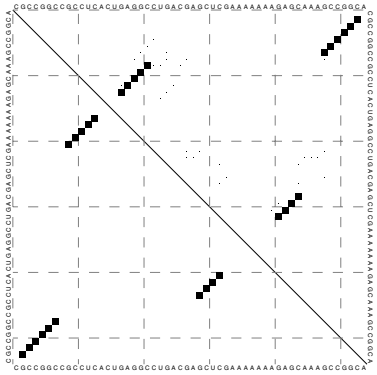


Figure 12.2.3 Using RNAinverse to design an artificial hammerhead ribozyme. (**A**) Command-line input, (**B**) A dot plot of the correct mfe structure with many alternative foldings, and (**C**) A dot plot of a sequence with an extremely well-defined structure.

the time needed for the search varies widely. For sequence lengths beyond 100 nt, searching for rare structures is of little use. If RNAinverse takes an extremely long time and produces unsatisfactory results, this may be because the target structure is too rare.

Small changes to the target structure are often sufficient to make the design problem tractable, the most common problem being isolated base pairs (i.e., helices of length 1). Except for very small structures, isolated base pairs should be avoided either by deleting the pair or elongating the helix.

4. Add sequence constraints.

Functional RNA molecules will often not only require a particular structure but will have to fulfill certain sequence constraints. For example, a binding site may require a hairpin with a particular loop sequence. Such sequence constraints can be specified by supplying a starting sequence as described in step 1. Any lowercase letters in the start sequence will remain unchanged during the optimization. One may supply an `N` *for any positions that should use a random character initially. When using sequence constraints for paired positions, make sure that the constrained sequence forms valid base pairs.*

INSTALLING THE VIENNA RNA PACKAGE

Necessary Resources

Hardware

A personal computer running Linux is recommended; a Unix workstation (e.g., from Sun, SGI, or IBM) or Macintosh under OS X may be used, but these platforms are less well tested. PCs with MS Windows require significant extra installation effort.

Software

Web browser

ANSI compliant C compiler and related tools (make utility, shell, header files)

Optional: Perl5 installation for compiling the Perl modules; some of the bundled sample scripts require the Perl/Tk module

While Linux and Unix typically provide all required tools in a default installation, Mac OS X does not come preinstalled with development tools. The development package can be downloaded (free of charge) from *http://www.apple.com*. For MS Windows, the easiest solution is to install the CygWin environment from *http://www.cygwin.com*. This is a rather large package that provides a complete GNU development environment running under Windows.

1. Point the Web browser to *http://www.tbi.univie.ac.at/~ivo/RNA/* and download the source code for the latest version (currently ViennaRNA-1.5). Save in a convenient location.
2. Unpack the tar file by running:

```
gunzip ViennaRNA-1.5.tar.gz
tar -xvf ViennaRNA-1.5.tar
```

3. To configure, build, and install the package, run:

```
cd ViennaRNA-1.5
./configure
make
make install
```

The `make install` *should be run as root if one is installing to the default location. This will install the main programs in* `/usr/local/bin`*; additional scripts and example programs will be installed in* `/usr/local/share/ViennaRNA/bin`*. To use these, one will have to either make sure the directory is in the* `PATH` *environment variable, or use the full path.*

The installation location can be controlled through options to the `configure` *scripts. For example., to install in the directory* `ViennaRNA` *in one's home directory, use:*

```
./configure –prefix=$HOME/ViennaRNA
```

For more detailed instructions see the `INSTALL` *file in the distribution or the documentation on the Web page.*

GUIDELINES FOR UNDERSTANDING RESULTS

Structure prediction is, of course, error-prone, and the user is ultimately forced to decide how much trust to put in the prediction. Several approaches described here can at least help with an informed decision.

In its simplest form, RNA structure prediction returns a single, optimal solution—the mfe structure. While this may be the most convenient form of structure prediction, a single predicted structure can give no hint as to the reliability of the prediction. The authors

therefore strongly recommend using base-pair probabilities and/or suboptimal folding to obtain an overview of plausible foldings and assess the how well defined a prediction has been obtained.

Current parameters are expected to predict some 70% of base pairs correctly (Mathews et al., 1999), on average, but these accuracies vary widely and may be below 40% in unfortunate cases. Measures of "well definedness" derived from base-pair probabilities or suboptimal structures can help identify such problematic cases, since well defined regions are usually predicted with much higher accuracy than less well defined regions.This is illustrated in Figure 12.2.1, which shows the prediction for a 5S rRNA. While the helix enclosing the multi-loop and the 3′ portion has a well defined structure, the first arm of the multi-loop shows several alternative structures and the optimal structure is indeed partly wrong in this region.

Of course, predictions that exhibit many structural alternatives need not be a consequence of inaccurate parameters, but may reflect real structural flexibility, which in turn may be of functional importance. Nevertheless, good predictions are harder to achieve in such cases.

Ideally, one should always strive to support predictions through experimental data. Alternatively, if several related sequences are available, sequence covariations can be used to support predicted structures. Even with only two or three sequences, the consensus structure predicted by RNAalifold will be much more accurate than prediction for single sequences.

COMMENTARY

Background Information

Single-stranded RNA molecules may fold back on themselves to produce double-helical regions interrupted by loops. A secondary structure is simply the list of base pairs thus formed. Here, the discussion is restricted to secondary structures without pseudo-knots, i.e., base pairs i, j and k, l with $i < k < j < l$ are not allowed.

RNA secondary structure prediction is based on a realistic energy model using experimentally determined parameters (Mathews et al., 1999). Because of the additivity of the energy model, a base pair divides a structure into two independent parts. This observation allows the construction of efficient dynamic programming to solve the folding problem.

These algorithms require CPU time that grows with the cube of the sequence length, while memory requirements grow quadratically. This means that sequences up to some 10,000 nt can be handled well on present-day workstations.

Dynamic programming algorithms are typically used to solve an optimization problem, producing a single optimal solution (here the mfe structure). The dynamic programming scheme can, however, be used just as well to compute ensemble quantities such as the partition function, or enumerate all structures within a range of the optimum.

Using more than a single optimal structure is particularly important for RNA structure prediction since the optimal structure problem is ill-conditioned: because the number of structures to be considered grows exponentially with sequence length, even small inaccuracies will eventually cause wrong structures to be predicted as optimal.

Furthermore, an RNA molecule at room temperature will fluctuate between different structures; while the thermodynamic ensemble is often dominated by the mfe structure, it may also contain very diverse structures of similar energy. All of these structures, not just the most stable one, can be functionally relevant.

Using the partition function Q, the frequency of any structure, s, in the thermodynamic equilibrium ensemble can be computed as its Boltzmann weight:

$$p = \frac{\exp\left(\frac{-E(s)}{RT}\right)}{Q}$$

or, using the ensemble free energy:

$$F = -RT \log Q$$

it can be computed as:

$$p = \exp\left(\frac{-\left(E(s) - F\right)}{RT}\right)$$

The difference between "minimum free energy" and "ensemble free energy" sometimes causes confusion. Each secondary structure is a macro state described by a free energy whose entropy stems from the fact that a secondary structure comprises many micro-states. The ensemble free energy also contains a different type of entropy related to the fact that the ensemble contains many secondary structures. Thus, the ensemble free energy F is always lower than the minimum free energy.

Critical Parameters and Troubleshooting

The energy rules used by the Vienna RNA programs can be tweaked using several parameters; here only a few are mentioned. For a complete list of options and their effect, see the main pages and the RNAlib documentation.

RNA secondary structures are affected by their environment through parameters such as temperature and ionic conditions. The `-T` option can be used to change the folding temperature. This is done by extrapolating the energy parameters to the desired temperature. The accuracy of the prediction, however, is expected to be highest at the default temperature of 37°C. Options to correct energy parameters for ionic conditions will hopefully be provided in an upcoming release.

Another interesting command line switch is the `-noLP` option. This will produce structures without isolated base pairs (e.g., helices of length *1*). Since such isolated base pairs are not well described by the energy model, this should not affect prediction accuracy, but leads to strong reduction of the number of structures in suboptimal folding.

Advanced Parameters

A sometimes confusing option in the Vienna RNA package pertains to the treatment of so-called dangling ends. Dangling ends are the unpaired bases adjacent to pairs in multi-loops and the external loop. They stabilize the structure through stacking interaction. The programs implement three different levels of dangling end treatment: Using the option `-d2` chooses a simplified dangling-end model that is however supported in all algorithms. `-d1` chooses the default treatment but is unavailable in the partition function algorithm (it will revert to `-d2`). Finally `-d3` enables limited support for coaxial stacking in multi-loops which may improve predictions (Walter et al., 1994), but is available only for mfe folding and energy evaluation. Using `-d2` ensures that all algorithm use the exact same energy model facilitating comparison.

The Vienna RNA package uses the standard energy parameters available from Doug Turner's group (*http://rna.chem.rochester.edu/*) and described in Mathews et al. (1999). Customized energy parameters can be supplied using the `-P` option. Files containing the current parameter set as well as an older version are included in the distribution. The authors hope to include DNA parameters as well, as soon as a complete and freely distributable parameter set becomes available.

Suggestions for Further Analysis

Comparing secondary structures: The RNAdistance program of the Vienna RNA package can be used to compare RNA secondary structures using various distance measures. A useful measure to compare alternative structures of the same sequence is the "base pair distance," which counts the number of pairs present in one structure but not the other. As distance measures that can be used to compare structures of different length, the program offers simple string alignment (on the bracket notation) and tree editing (Shapiro and Zhang, 1990).

Calculating specific heat curves: The melting behavior of an RNA molecule is best described by its specific heat curve. The RNAheat program calculates the specific heat of an RNA sequence as a function of temperature by numerical differentiation from the partition function. The output is a list of coordinates suitable for a *x-y* plotting tool such as xmgrace. Peaks in the specific heat curves mark structural transitions, the highest temperature transition being the final unfolding of the molecule.

Finding conserved RNA secondary structures: The alidot program (Hofacker et al., 1998; Hofacker and Stadler, 1999) can be used as an alternative to RNAalifold. Instead of intermixing folding and covariance analysis, alidot uses structure predictions for individual sequences that are then combined with a sequence alignment to find conserved structural motifs. This approach may be preferable for

long sequences with interspersed conserved region, but without an overall consensus secondary structure.

Folding dynamics: RNA folding kinetics sometimes play an important role for RNA functions. The barriers program available from *http://www.tbi.univie.ac.at/~ivo/RNA/Barriers/* can be used in conjunction with RNAsubopt to analyze the energy landscape of an RNA molecule in terms of local minima and energy barriers (Flamm et al., 2002). The kinfold program (Flamm et al., 2000) can be used to do explicit simulations of RNA folding dynamics; it is available from *http://www.tbi.univie.ac.at/~xtof/RNA/Kinfold/*.

Literature Cited

Flamm, C., Fontana, W., Hofacker, I.L., and Schuster, P. 2000. RNA folding at elementary step resolution. *RNA* 6:325-338.

Flamm, C., Hofacker, I.L., Stadler, P.F., and Wolfinger, M.T. 2002. Barrier trees of degenerate landscapes. *Z. Phys. Chem.* 216:155-173.

Hofacker, I.L. and Stadler, P.F. 1999. Automatic detection of conserved base pairing patterns in RNA virus genomes. *Comput. Chem.* 23:401-414.

Hofacker, I.L., Fontana, W., Stadler, P.F., Bonhoffer, S., Tacker, M., and Schuster, P. 1994. Fast folding and comparison of RNA secondary structures (the Vienna RNA Package). *Monath. Chem.* 125:167-188.

Hofacker, I.L., Fekete, M., Flamm, C., Huynen, M.A., Rauscher, S., Stolorz, P.E., and Stadler, P.F. 1998. Automatic detection of conserved RNA structure elements in complete RNA virus genomes. *Nucl. Acids Res.* 26:3825-3836.

Hofacker, I.L., Fekete, M., and Stadler, P.F. 2002. Secondary structure prediction for aligned RNA sequences. *J. Mol. Biol.* 319:1059-1066.

Mathews, D., Sabina, J., Zucker, M., and Turner, H. 1999. Expanded sequence dependence of thermodynamic parameters provides robust prediction of RNA secondary structure. *J. Mol. Biol.* 288:911-940.

McCaskill, J. 1990. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers* 29:1105-1119.

Schuster, P., Fontana, W., Stadler, P.F., and Hofacker, I.L. 1994. From sequences to shapes and back: A case study in RNA secondary structures. *Proc. R. Soc. London B Biol. Sci.* 255:279-284.

Shapiro, B. and Zhang, K. 1990. Comparing multiple RNA secondary structures using tree comparisons. *Comput. Appl. Biosci.* 6:309-318.

Walter, A.E., Turner, D.H., Kim, J., Lyttle, M.H., Muller, P., Mathews, D.H., and Zuker, M. 1994. Coaxial stacking of helixes enhances binding of oligoribonucleotides and improves predictions of RNA folding. *Proc. Natl. Acad. Sci. U.S.A.* 91:9218-9222.

Wuchty, S., Fontana, W., Hofacker, I.L., and Schuster, P. 1999. Complete suboptimal folding of RNA and the stability of secondary structures. *Biopolymers* 49:145-165.

Zuker, M. 1989. On finding all suboptimal foldings of an RNA molecule. *Science* 244:48-52.

Zuker, M. and Stiegler, P. 1981. Optimal computer folding of larger RNA sequences using thermodynamics and auxiliary information. *Nucl. Acids Res.* 9:133-148.

Key References

Hofacker, I.L. 2003. The Vienna RNA secondary structure server. *Nucl. Acids Res.* 31: 3429-3431.

Describes how to perform several of the functions discussed in this unit on the Web using the Vienna RNA server.

Hofacker et al., 1994. See above.

The paper describing the first release of the Vienna RNA package, including a description of the underlying algorithm.

Internet Resources

http://www.tbi.univie.ac.at/~ivo/RNA/

Site at which to download the latest version of the Vienna RNA package and read online manuals.

http://rna.tbi.univie.ac.at/

Web interfaces to several of the Vienna RNA programs.

Contributed by Ivo L. Hofacker
University of Vienna
Vienna, Austria