A Critical Review of Computational Methods for RNA Secondary Structure Prediction

Adam Silverman
Biochem218
Submitted June 6, 2003

Introduction

The three-dimensional structure of RNA molecules is crucial to their function. The primary structure is determined by the sequence of G, A, C, and U bases in a strand. Secondary structure consists of hydrogen-bonded base pairings between complimentary bases (G and C or A and U typically, or G and U non-canonically) and the loops formed by unpaired bases. Secondary structure is determined discretely: each base is either paired or not. An example of an RNA secondary structure is shown in Figure 1. Tertiary structure is made up of interactions between secondary structures, generally through formation of additional hydrogen bonds or hydrophobic interactions. The interactions that determine secondary structure are generally significantly stronger than those governing tertiary structure because. There are no continuously varying parameters such as bond lengths, angles, or interatomic distances, which must be accounted for in tertiary structure. It is generally assumed that the influence of tertiary structure on secondary structure is negligible; consequentially, secondary structures can be determined independently of tertiary structures.

Myriad algorithms have been developed for the prediction of RNA secondary structure from its primary sequence. In theory, the number of valid secondary structures for a given sequence is greater than 1.8^N, where N is the number of nucleotides (Zuker, *et a*l., 1991). Most folding programs fit into one or more of four classes: (1) “Basic" algorithms predict hairpin and simple loop formation, but they exclude the prediction of multibranched loops and perform very basic energy minimization. The first algorithms written were of this type, and most have been updated or are no longer in use. (2) "Combinatorial" methods generate lists of all possible secondary structure elements and piece them together in all possible ways to find those with the lowest free energy. (3) "Recursive" algorithms build the secondary structure one nucleotide at a time while computing minimum energies along the way. Dynamic programs, which employ recursive algorithms, compute folding in time based on low energy paths of achieving secondary structure. (4) “Comparative sequence analysis" algorithms find conserved structure for a set of sequences using stochastic optimization on a population of tentative solutions.

Despite their success, current secondary structure methods tend to have problems in several areas. Most significantly, many different foldings are possible near the energy minimum, and it is difficult or impossible to determine which of these "suboptimal" folds is correct. For example, for the 5.8S RNA from *C. cohnii*, the minimum energy folding

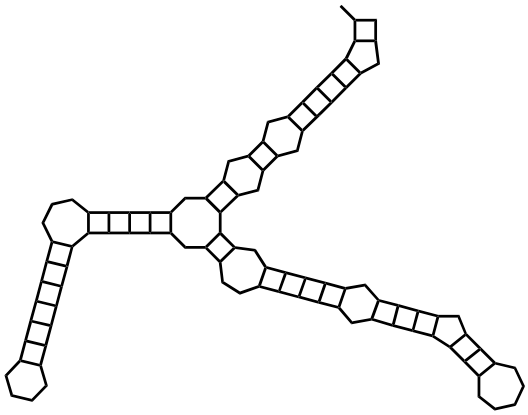

Figure 1. Folding of a sample RNA. Several different loop types occur and are designated by letters: B = bulge loop; I = interior loop; H = hairpin loop; M = multibranched loop. Unmarked areas are stacking regions between base hydrogen bonded base pairs.

and an alternative folding with less than 5% energy difference do not have a single base pair in common (Zuker, 1989b). Secondly, the energy minimization rules are derived from melting data on small oligonucleotides, and may not be completely accurate for large RNAs. Any algorithm that relies on free energy minimization to achieve an optimal structure is only as good as the thermodynamic parameters used, and in some cases the values may not be totally reliable. Furthermore, most current algorithms assume that the total free energy of an RNA secondary structure can be computed by summing the contributions of the components, but this may not be accurate in many cases.

Free Energy Minimization

Most RNA secondary structure prediction algorithms perform thermodynamic optimization on a series of plausible structures in order to obtain the structure or structures with the lowest equilibrium free energy. Basic thermodynamic principles indicate that the structure lowest in free energy should be the most stable and barring outside influences, the correct fold. Unfortunately, not all of the factors that determine the energy of a fold are understood, and computational time limitations would make it infeasible to include all of such influences. Current methods ignore free energy contributions from tertiary structure, a reasonable assumption because the forces determining tertiary structure are weaker than those governing secondary. However, tertiary structure contributions may play an increasing role as the length of the RNA strand is increased, because more complex folds are possible. The values for such contributions have not been determined.

The free energy of a secondary structure is determined by summing the energy contributions of all base pairs, loops, hairpins, etc. Energy contributions have been shown to be additive for short oligonucleotides in melting studies, but the free energies for longer RNA strands (>50-100 nucleotides) have not been empirically determined. Values for contributions of individual secondary structure elements were determined by melting studies with short oligonucleotides (Freier, *et al*., 1986). For simple base-pairing energies, the "individual nearest neighbor" (INN) method has long been used, but has been updated recently to improve the accuracy of values (Xia, *et al*., 1998). The nearest neighbor approach assumes that the thermodynamic stability of a base pair is solely

dependent on the identity of adjacent bases. Thus, thermodynamic contributions from both base pairing and base stacking are considered.

Thermodynamic properties were obtained by plotting melting data from short RNA duplexes (4-10 base pairs). Thermodynamic values are related to the melting data by the equations below:

$$K = \exp(-\Delta G^\circ(T)/RT) = \exp\left(-\frac{\Delta H^\circ}{RT} + \frac{\Delta S^\circ}{R}\right) = \frac{\alpha}{2(C_T/a)(1-\alpha)^2}$$

$$T_M^{-1} = \frac{R}{\Delta H^\circ}\ln(C_T/a) + \frac{\Delta S^\circ}{\Delta H^\circ}$$

α is the fraction of total single strand in the duplex as a function of temperature, and *a* is a constant equal to1 for self-complimentary strands or 4 for non-self-complimentary strands. Thermodynamic parameters were derived from these experiements (see Table 1). For repeated experiments, the measured values for ΔG^o, ΔH^o, and ΔS^o for each nearest neighbor set are predicted within 3.2%, 6.0%, and 6.8, respectively (Xia, *et al*., 1998).

Table 1. RNA Thermodynamic Parameters for Nearest - Neighbor Model, 1 M NaCl, pH 7[a] (Xia,*et al*., 1998)

parameters	ΔG°_{37} (kcal/mol)	ΔH° (kcal/mol)	ΔS° [b] (eu)
5′AA3′ 3′UU5′	−0.93 (0.03)	−6.82 (0.79)	−19.0 (2.5)
5′AU3′ 3′UA5′	−1.10 (0.08)	−9.38 (1.68)	−26.7 (5.2)
5′UA3′ 3′AU5′	−1.33 (0.09)	−7.69 (2.02)	−20.5 (6.3)
5′CU3′ 3′GA5′	−2.08 (0.06)	−10.48 (1.24)	−27.1 (3.8)
5′CA3′ 3′GU5′	−2.11 (0.07)	−10.44 (1.28)	−26.9 (3.9)
5′GU3′ 3′CA5′	−2.24 (0.06)	−11.40 (1.23)	−29.5 (3.9)
5′GA3′ 3′CU5′	−2.35 (0.06)	−12.44 (1.20)	−32.5 (3.7)
5′CG3′ 3′GC5′	−2.36 (0.09)	−10.64 (1.65)	−26.7 (5.0)
5′GG3′ 3′CC5′	−3.26 (0.07)	−13.39 (1.24)	−32.7 (3.8)
5′GC3′ 3′CG5′	−3.42 (0.08)	−14.88 (1.58)	−36.9 (4.9)
initiation[c]	4.09 (0.22)	3.61 (4.12)	−1.5 (12.7)
per terminal AU[d]	0.45[d] (0.04)	3.72[d] (0.83)	10.5[d] (2.6)
symmetry correction (self-complementary)	0.43	0	−1.4
symmetry correction (non-self-complementary)	0	0	0

[a] Numbers in parentheses are uncertainties for parameters. [b] Calculated from nearest-neighbor parameters for ΔG°_{37} and ΔH° (see Materials and Methods). [c] Includes potential GC end effects. [d] Parameter

Theromodynamic parameters for loops were determined analogously from additional melting data. The stability of hairpins, bulges, and other loops is largely dependent on four factors: (1) sequence of the loop, (2) nucleotides adjacent to and closing the loop, (3) nearby sequences not adjacent to the loop, and (4) size and shape of the loop (Serra and Turner, 1995; Longfellow, *et al*., 1990; Giese, *et al*., 1998; Serra, *et al*., 1997). The accuracy of the melting data when applied to very large loops is unknown; for example, widely used thermodynamic parameters for hairpins come from melting studies on hairpins of only six nucleotides (Serra, *et al*., 1994), while hairpins of over 50 nucleotides are known. Data have suggested that very small changes in energy parameters often result in very large changes in predicted folding (Zuker, et al., 1991). The thermodynamic values determined in these studies may provide sufficient approximations; however, better size- and sequence-based parameters would be extremely useful in determining suboptimal structures which are very close to the lowest energy structure.

Determination of Suboptimal Folds: Combinatorial and Recursive Algorithms

Combinatorial and recursive folding algorithms are capable of finding minimum energy secondary structures, while comparative sequence analysis algorithms find phylogenically conserved structures. The combinatorial method forms structures by combining all possible helices in all possible ways, thereby predicting a series of folds (Dumas and Ninio 1982). Unfortunately, programs like this are extremely time consuming and require a great deal of computer memory, since the number of potential folds increases exponentially with the length of the sequence. Most combinatorial programs are limited to folding about 200 nucleotides. Structures with energy near the minimum can be reported, but in many cases too many folds within a reasonable threshold are available, making it difficult to make any statistical or biological sense of them.

Recursive algorithms work in two stages. The first part, known as the "fill", starts with small fragments (usually pentanucleotides) and builds up to larger segments in a recursive fashion by iteratively minimizing the free energy. Ultimately, the fill computes and stores minimum folding energies for all fragments of the sequence. Next, the "traceback" computes a minimum energy structure by searching through the matrix of stored energies and combinig compatible fragments. Recursive algorithms can be much faster than combinatorial algorithms; at best they may determine secondary structures in time proportional to the cube of the sequence length (Zuker, 1989b).

Unfortunately, the process of recursively optimizing RNA secondary structures only allows determination of one optimized structure. Researchers have attempted to deal with this problem in a number of ways. One method that at first seems reasonable is to perturb the thermodynamic parameters. However, the suboptimal folds generated would be highly sensitive to the design of the perturbation algorithm rather than having any real statistical significance. Another possibility is to take a standard recursive algorithm and set a threshold energy level in order to output all structures with energies below the threshold. However, if the threshold is set too low not much variation is possible, and if it is set too high, too many structures may be generated for reasonable evaluation. For a sequence of about 400 nucleotides, a structure that is about 80% correct can be found from a group of about 20 structures within 5% of the lowest free energy structure, while the single best structure is generally within 2% of the free energy of the optimal structure (Zuker, et al., 1991).

The first successful recursive suboptimal folding algorithm was designed by applying observations about circular RNA to linear RNA (Zuker, 1989b). The choice of an origin is arbitrary in circular RNA, so in a circular RNA composed of ribonucleotides r_1, $r_2 \ldots r_n$, a base pair linking r_i and r_j divides the secondary structure into two segments: from r_i to r_j and from r_j to the origin r_i. It is apparent that a recursive algorithm could find many folds for circular RNA simply by starting at different origins. This principle can be generalized to linear RNA simply by considering the first and last bases to be adjacent and allowing them to pair with each other if necessary.

The minimum free energies of the two segments, V(i,j) and V(j,i), can be added to determine the minimum energy for a structure containing the r_i-r_j base pair. The

minimum value of V(i,j) + V(j,i) across all possible base pairs r_i-r_j is the minimum folding energy, E_{min}. To obtain suboptimal folds, the algorithm looks for base pairs for which V(i,j) + V(j,i) is close to E_{min}. Rather than simply choosing structures within a fixed value of E_{min}, however, the algorithm generates optimal and suboptimal structures by choosing an optimal or suboptimal base pair (a base pair that fits certain probabilistic criteria, described in detail in Zuker, 1989b), and computing the best folding for that base pair. The result is that not all possible structures need to be computed, which speeds up computational time compared to combinatorial programs. Structures with at least 5 to 10% variation from the minimum energy structure are determined.

Comparative Sequence Analysis Algorithms

The folds of structural RNAs (tRNAs and rRNAs) are highly conserved among all kingdoms of life and have been widely used to determine phylogenic relationships between different species (Kumar and Rzhetsky, 1996). One way to predict the fold of structural RNA's is through phylogenic-comparative analysis. Most algorithms of this type rely on an analysis of aligned nucleotide sequences to determine conserved regions of secondary structure.

This approach is governed by the assumption that mutations that disrupt Watson-Crick base pairs have a negative effect which may be overcome by a second compensatory mutation in the other half of the stem, restoring the base pair. This sort of evolution results in a pattern of nucleotide substitutions, called covariations, that can be detected in sequence alignments of homologous RNA sequences from different species. A covarying site is one that may differ between species but maintains its potential to form a base pair (e.g., GC in one species replaced by AU in another).

A major problem with early comparative sequence analysis algorithms was that phylogenic relationships of the aligned sequences and levels of sequence divergence were not considered. One way this problem has been overcome is by generating a pairing parameter □ which measures the pattern of nucleotide substitution at paired sites verus those at unpaired sites in order to make quantitative comparisons in evolutionary conservation (Muse, 1995) . This "likelihood-ratio test" (LRT) approach has the severe drawback that its statistical significance is questionable for helices less than 10 base pairs in length (Parsch, et al., 2000). Numerical and probabilistic models can be made to help overcome this problem, however. In Parsch's algorithm, a complete list of potential RNA helices, along with their □values, are generated. Compatible helices are then grouped into subsets, which are combined to form potential secondary structure models. For each set of helices, a total □ value is determined by summing the □ values for each individual sequence; the optimal structure is the one with the greatest total □ value.

While comparative sequence analysis alone can be successful in finding regions of conserved structure among RNA sequences, these methods tend not to be globally accurate because no energy minimization is performed. There is a trade-off between the number of sequences entered and the accuracy of the results. Inputting more sequences will yield fewer regions of conservation, but these regions will tend to be more accurate;

inputting fewer sequences will give a greater number of conserved regions with lower accuracy. In general, the reliability of non-conserved regions is questionable. Furthermore, many comparative sequence analysis algorithms do not allow non-canonical base pairs. Algorithms that do allow GU wobble pairs or mismatches generally use a weighted penalty for structures containing mismatches, but without considering any thermodynamic implications, structures may be generated that lack biological significance.

A Genetic Algorithm with Energy Minimization

Chen and coworkers have developed one a comparative sequence analysis algorithm that uses a very different approach to find common RNA secondary structures for a set of RNA sequences (Chen, et al., 2000). Their method is a "genetic algorithm", which is intended to mimic genetic evolution. Genetic algorithms operate on a population of tentative solutions, each of which has an encoded representation equivalent to the genetic material of an individual in nature. The solutions are modified by mutation (random changes) and crossover (recombination of features), and the modified solutions are selected by predefined fitness criteria, energy minimization in this case.

Unlike the previously discussed comparative sequence analysis methods, Chen's method does not require an alignment to determine a common structure for a series of RNA sequences. Both the structural energy and structural similarity among sequences of potential solutions are considered. Free energy is minimized by the nearest neighbor approach, with penalties or bonuses for other secondary structure elements (since the focus is on structural similarity, the free energy rules are not as complex as those for recursive algorithms, see below).

The genetic algorithm proceeds as follows. For each sequence, an initial population of n structures is generated. Crossover, mutation, and selection are iterated with free energy as the fitness criterion, until the stability criteria of the structures are reached. For each sequence, a conservation score $cons(T_p)$ is evaluated for each structure T_p in the current generation of a sequence S_p; stem scores $cons(s_i)$ are computed for each stem s_i in each structure T_p. Mutations and crossovers are then performed on the current generation. A total of $3n$ structures that satisfy $cons(T) > h_c$ and $e(T) < e_c$, where h_c is a conservation parameter and $e(T)$ is the free energy of structure T, are collected from this iteration. The next generation for each sequence is then selected for by forming a set F from the unique $3n$ structures. A distance function d_i is calculated for each structure T_i in F. The distance function is defined as $d_i = \sum_j d_{ij}$, where d_{ij}, the distance score between structure T_i and T_j, is defined by $d_{ij} = 1 - n_{ij}/m_{ij}$, where n_{ij} is the number of base pairs in common between the two solutions and m_{ij} is the maximum number of base pairs of the two structures. The structures are then sorted by ascending values of $sc(i) = (best\_fit - cons(T_i)/d_i$, and the top n structures are selected for the next iteration. After the maximum number of generations and the structures begin to converge, a structure T' is eliminated if: (1) T' is a substructure of some other structure T; (2) $e(T') < e(T)$; (3) $cons(T') < cons(T)$. In this way the most conserved structure with the lowest energy is generated. It is also possible to consider suboptimal structures.

This approach has been very successful for determining conserved structures of tRNAs and small rRNA subunits. It does have some major limitations, however. As it is designed for searching structural space for folds that satisfy conditions of structural conservation and thermodynamic stability, it can only be applied to phylogenicly related sequences. Because the algorithm applies conservation criteria before fitness criteria to throw out structures, low energy structures that are not highly conserved in early iterations might be missed. The authors are also unclear as to whether non-structural RNAs have enough evolutionary similarity to be candidates for this program, so one must assume that it is limited to tRNA and rRNA.

The computational time is proportional to $n^2m^2N^2$, where n is the maximum number of stems, N is the number of sequences, and m is the maximum number of structures among the N sequences. This program is significantly more time-intensive than many others, including MFOLD (see below).

Structure Optimization by Energy Minimization: The MFOLD Algorithm

Unlike comparative sequence analysis algorithms, which find common structures for a group of sequences, recursive algorithms find optimal structures for a single sequence. These programs typically rely exclusively on free energy minimization for determining the best structures, so the thermodynamic parameters used are of critical importance.

Early recursive algorithms were problematic in many regards. The slow speed of computers in the early 1980s was perhaps the biggest setback, but thermodynamic parameters were not very accurate, nor were they always correctly incorporated into algorithms. Thermodynamic data incorporated into the first RNA secondary structure prediction algorithms was thought to have an uncertainty of ±0.2 to 0.5 kcal for base-paired helical regions and ±1 to 2 kcal for loops (Williams and Tinoco, 1986). In some programs, the free energy minimization was designed to make the program obtain results consistent with experiments (Zuker and Stiegler, 1981).

The MFOLD algorithm is recursive and based on free energy minimization. It is capable of producing suboptimal structures (Walter, *et al*., 1994). The energy minimization algorithm used in the current version of MFOLD is described in detail (Mathews, *et al*., 1999). It assumes that RNA thermodynamics has a linear dependence on the frequency of base pair doublets, that is, $\Delta G^o(\text{duplex}) = \Delta G^o_{\text{init}} + \Delta n_j \Delta G^o_j(\text{NN}) + m_{\text{term-AU}} \Delta G^o_{\text{term-AU}} + \Delta G_{\text{sym}}$. The $\Delta G^o_j(\text{NN})$ terms are the greatest contributors, being the free energy contribution of the j^{th} nearest neighbor with n_j occurrences in the sequence. ΔG^o_{init} is the translational and rotational energy loss for converting two molecules into one in forming the first base pair in a sequence. It is assumed to be independent of the length of the sequence. The $m_{\text{term-AU}} \Delta G^o_{\text{term-AU}}$ term is a correction factor to account for the fact that AU base pairs are weaker than GC pairs when at the terminus of a sequence of paired bases. The ΔG_{sym} term comes from the 2-fold rotational symmetry present in self-complimentary duplexes, but is equal to zero in non-self-complimentary duplexes.

The contribution of hairpin loops of less than 3 unpaired bases is determined by subtracting the stabilities of the stems, as calculated by nearest neighbor values. For hairpins with more than 3 unpaired bases (n>3), the energy contribution is approximated from the loop length and the sequences of the closing base pair and first mismatch by the equation: $\Delta G^{o}_{loop}(n>3) = \Delta G^{o}_{init}(n) + \Delta G^{o}$(stacking of first mismatch) + ΔG^{o}_{bonus}(UU or GA first mismatch) + ΔG^{o}_{bonus}(GU closure) + $\Delta G^{o}_{penalty}$(oligo-C loops). The values for these parameters are empirically computed constants.

Bulge loops, internal loops, and multibranched loops are destabilizing to RNA structure. The stability of bulges is computed by $\Delta G^{o}_{bulge} = \Delta G^{o}_{init}(n) + \Delta G^{o}_{bp\ stack}$(bulges of one nt only). Values $\Delta G^{o}_{init}(n<3)$, where n is the number of unpaired nucleotides in the bulge, have been experimentally determined. For n = 4, 5, or 6, the free energy is linearly increased, and for n>6, it is approximated by $\Delta G^{o}_{init}(n>6) = \Delta G^{o}_{init}(6) + 1.75$ RT ln (n/6). An approximation is used to model the free energy of internal loops (Serra and Turner, 1995): $\Delta G^{o}_{loop} = \Delta G^{o}_{init}(n1 + n2) + \Delta G^{o}_{asymm}|n1 - n2| + \Delta G^{o}_{AU/GU\ closure\ penalty} + \Delta G^{o}_{UU/GA/AG\ bonus}$, where n1 is the number of nucleotides on one side of the loop and n2 is the number of nucleotides on the other. Again, values for these parameters were obtained from melting studies (Peritz, *et al*., 1991). The stability of multibranched loops is approximated by the equation: $\Delta G^{o}_{loop} = a_1 + b_1 n + c_1 h + \Delta G^{o}_{dangle}$, where n is the number of unpaired nucleotides, h is the number of branching helices, and a_1, b_1, and c_1 are constants. Multibranched loops may have unfavorable contributions from initiation and favorable contributions from stacking, particularly in cases where coaxial stacking may exist. Consequentially, additional correctional terms may be included.

Many recursive algorithms have a similar general structure. How algorithms handle thermodynamic parameters will determine the accuracy of the structures generated. Secondary structure prediction programs can only be as accurate as the thermodynamic parameters allow them to be, so better thermodynamic values will generate better folds. The MFOLD algorithm uses the best thermodynamic data that is available, but it is still limited by data from a small number of experiments.

Free Energy Minimization and Comparative Sequence Analysis in One: The Dynalign algorithm

The most reliable method for determining the conserved structure of a series of RNA sequences is to combine comparative sequence analysis with free energy minimization. The Dynalign algorithm is one of the most successful for determining a common structure for two phylogenically related sequences (Mathews and Turner, 2002). It is a dynamic algorithm that aligns two sequences and finds a common structure. It uses thermodynamic parameters described above for the MFOLD algorithm for prediction of free energies (Mathews, *et al*., 1999). This program is perhaps the most accurate written to date; for tRNAs and 5S rRNAs, Dynalign predicted 86.1% and 86.4% of known base pairs, compared to 59.7% and 47.8% for free energy minimization alone.

Two phylogenically related sequences are input, and a sequence alignment and common structure are output. Base pairs are only permitted in the common structure if both

sequences allow a base pair at the position, with one exception: a single inserted base pair may be included in one structure if it is between two conserved base pairs. The free energy is minimized by $\Delta G^{o}_{tot} = \Delta G^{o}_{seq1} + \Delta G^{o}_{seq2} + (\Delta G^{o}_{gap})(\#\text{ of gaps})$. Gaps are locations in the sequence alignment where a nucleotide in one sequence has no analog in the other sequence. ΔG^{o}_{gap} is an empirical factor that penalizes gaps in the alignment; the value of ΔG^{o}_{tot} itself does not depend on matching nucleotides in the alignment, so no sequence identity is actually required for structure prediction. In fact, sequence identity is not explicitly scored. Instead it is implicitly considered in the free energy nearest neighbor parameters.

The algorithm is a four-dimensional dynamic program divided into fill and traceback steps. The fill steps calculate three free energy arrays, W(*i*,*j*,*k*,*l*), V(*i*,*j*,*k*,*l*), and W5(*i*,*k*). W(*i*,*j*,*k*,*l*) is the sum of the minimum free energies for nucleotide fragments *i* to *j* from the first sequence and *k* to *l* from the second sequence with *i* aligned to *k* and *j* aligned to *l*. V(*i*,*j*,*k*,*l*) is the same as W(*i*,*j*,*k*,*l*), except that *i* is base- paired with *j*, and *k* is based paired with *l*. W5(*i*,*k*) is the sum of free energies of nucleotide fragments from 1 to *i* in the first sequence and 1 to *k* in the second. An array W5(N_1,N_2), where N_1 and N_2 are the lengths of sequences 1 and 2 respectively, is the lowest free energy sum for a structure common to both sequences. When these arrays are calculated, the structural conformations that satisfy the minimal free energies are not explicitly calculated. The traceback steps use the information in the energy arrays to find the conserved structure that has the lowest free energy.

This approach has a number of advantages over genetic algorithms or algorithms based on free energy minimization alone. Though it combines many of the advantages of these methods and eliminates some of the shortcomings, Dynalign is not without its own limitations. Like algorithms based on free energy minimization alone, the ultimate accuracy of Dynalign's structures is dependent on the quality of the thermodynamic parameters. Unlike genetic algorithms, Dynalign is unable to align and determine the structure for more than two common sequences. This is because the dimensionality of the energy functions is equal to the square of the number of sequences being aligned. The algorithm finds the common structure of two sequences using four-dimensional energy functions; to find the structure for three or four sequences it would need to use nine- or sixteen-dimensional functions. It is unlikely that computers will be fast enough or have enough memory to handle this kind of data any time soon.

Improving prediction algorithms

The best current RNA secondary structure prediction algorithms are around 80% accurate for structural RNAs with known structural information. There are many ways that current algorithms could be improved. I would suggest two very generic improvements that could have very positive impacts on most algorithm types. The issue of accuracy of thermodynamic parameters for free energy minimization algorithms has already been discussed extensively. As mentioned, current thermodynamic parameters were obtained by performing melting studies on small oligonucleotides, but the stabilizing or destabilizing effects of longer sequences are unknown. For example, it is unknown

whether there is inherent additional stability for a hairpin containing a 20 base pair stem versus a 5 base pair stem. Modern automated synthesis methods have made it possible to synthesize fairly long (>50 nucleotide) RNAs in modest yields, so experiments of this sort are easily performed. More extensive melting studies, especially on longer loops, to determine additional stabilization values for hairpins and destabilization values for bulges and internal loops, could be very helpful to improve current algorithms. An algorithm that could iteratively consider the effects of increasing loop size in a sequence-specific manner would have a great deal of utility.

Secondly, most current algorithms do not allow input of information obtained from experimental data. For example, biochemical experiments may show that certain sets of bases in an RNA are paired. Rather than simply using this information to confirm the structure, this information could be used to bias the structure. Such restraints could drastically reduce computational time, since the number of possible structures would be greatly reduced. Analogously, a useful additional feature on algorithms that use phylogenic information to determine common structures would be to allow the user to input phylogenic restraints based on known relationships. Even better, the algorithms could be constructed with a database of known phylogenies. This could also drastically reduce computational time, though it runs the risk of missing unexpected relationships.

References

Chen, J.-H.; Le, S.-Y.; Maizel, J. V. *Nuc. Acid. Res*. **2000**, *28*, 991-999.

Dumas, J. –P.; Ninio, J. *Nuc. Acid. Res*. **1982**, *10*, 197-203.

Freier, S. M.; Kierzek, R.; Jaeger, J. A.; Sugimoto, N.; Caruthers, M. H.; Neilson, T.; Turner, D. *Proc. Natl. Acad. Sci*. **1986**, *83*, 9373-9377.

Giese, M. R.; Betschary, K.; Dale, T.; Riley, C. K.; Rowan, C.; Sprouse, K. J.; Serra, M. J. *Biochemistry* **1998**, *37*, 1094-1100.

James, B. D.; Olsen, G. J.; Lie, J.; Pace, N. R. *Cell* **1988**, *52*, 19-26.

Kumar, S.; Rzhetsky, A. *J. Mol. Evol*. **1996**, *42*, 183-193.

Longfellow, C. E.; Kierzek, R.; Turner, D. H. *Biochemistry*, **1990**, *29*, 278-285.

Mathews, D. H.; Sabina, J.; Zuker, M. Turner, D. H. *J. Mol. Biol*. **1999**, *288*, 911-940.

Mathews, D. H.; Turner, D. H. *J. Mol. Biol*. **2002**, *317*, 191-203.

Muse, S. V. *Genetics* **1995**, *139*, 1429-1439.

Parsch, J.; Braverman, J. M.; Stehpan, W. *Genetics* **2000**, *154*, 909-921.

Peritz, A. E.; Kierzek, R.; Sugimoto, N.; Turner, D. H. *Biochemistry* **1991**, *30*, 6428-6436.

Serra, M. J.; Axenson, T. J.; Turner, D. H. *Biochemistry* **1994**, *33*, 14289-14296.

Serra, M. J.; Barnes, T. W.; Betschart, K.; Gutierrez, M. J.; Sprouse, K. J.; Riley, C. K.; Stewart, L.; Temel, R. E. *Biochemistry* **1997**, *36*, 4844-4851.

Serra, M. J.; Turner, D. H. *Methods Enzymol.* **1995**, *259*, 242-261.

Walter, A. E.; Turner, D. E.; Kim, J.; Lyttle, M. H.; Muller, P.; Mathews, D. H.; Zuker, M. *Proc. Natl. Acad. Sci.* **1994**, *91*, 9218-9222.

Williams, A. L., Jr.; Tinoco, I., Jr. *Nuc. Acid. Res*. **1986**, *14*, 299-315.

Xia, T.; SantaLucia, J.; Burkard, M. E.; Kierzek, R.; Schroeder, S. J.; Jiao, X.; Cox, C.; Turner, D. H. *Biochemistry* **1998**, *37*, 14719-14735.

Zuker, M. *Methods Enzymol*. **1989a**, *180*, 262-288.

Zuker, M. *Science* **1989b**, *244*, 48-52.

Zuker, M.; Jaeger, J. A.; Turner, D. H. *Nuc. Acid. Res*. **1991**, *19*, 2707-2714.

Zuker, M.; Stiegler, P. *Nuc. Acid Res*. **1981**, *9*, 133-148.