Volume 16 Number 5 1988 Nucleic Acids Research

An RNA secondary structure workbench

Hugo M.Martinez

PO Box 0448, Department of Biochemistry and Biophysics, University of California, San Francisco, CA 94143, USA

Received August 17, 1987; Revised and Accepted November 23, 1987

ABSTRACT

A multiple approach to the study of RNA secondary structure is described which provides for the independent drawing of structures using base-pairing lists, for the generation of local structures in the form of hairpins, and for the generation of global structures by both Monte Carlo and dynamic programming methodologies. User-adjustable parameters provide for limiting the size of hairpin loops, bulges and inner loops, and constraints can be imposed relative to position-dependent base pairing.

INTRODUCTION

The program RNAFOLD has evolved from a number of considerations of how best to go about investigating RNA secondary structure. Experience in my computation laboratory (UCSFBCL) indicates that because the theoretical foundations of the current prediction methodology is far from satisfactory it is best to offer a spectrum of approaches. Hence the concept of a workbench in the form of four basic programs which constitute RNAFOLD.

The intent of these four programs is to provide a convenient and flexible environment. Thus, a user may simply wish to draw a secondary structure with base-pairing information derived from evolutionary considerations and/or biochemical tests. The drawing may be for publication purposes or for added insight as to potential base-pairing not previously considered. Hence the program DRAWSTRUCT. On the other hand, the requirement may be for structures derived from free energy considerations alone, in which case the programs STEMS, MONTECARLO and DYNPRO can be invoked according as the concern is with local or global structures. The program STEMS generates hairpins under specified constraints regarding the size of bulges, inner loops and hairpin loops. The program MONTECARLO generates a population of global structures based on a Monte Carlo method of folding previously reported on, Martinez [1], and the program DYNPRO generates a global structure based on dynamic programming methodology. Both MONTECARLO and DYNPRO can be subjected to base-pairing constraints of the type which exclude or favor specific, position dependent base pairs.

DRAWSTRUCT

A number of secondary structure drawing programs have been reported on [2]. Perhaps the most significant feature among these programs is the ability to handle the problem of overlaps, that is, the obscuring of one part of the drawing by another part. I have chosen the intermediate solution to this problem in that while the program makes an attempt to remove some of the overlapping when it occurs, the user can resolve remaining conflicts by interactively rotating and stretching stems. Fig 1A shows a drawing with some overlap and Fig 1B is the same drawing with overlaps interactively removed. The sequence involved is named 'myc3' and is the third exon from a chicken oncogene. This sequence is used in all the subsequent examples. The structure in the drawing was obtained as a folding using the MONTECARLO approach to be described below.

In order to have the feature of user-modifiability, the drawing consists of identifiable hairpins (stem + hairpin loop). Each hairpin has a unique number drawn in its loop. Selecting a hairpin by its number label, the user can rotate or stretch it relative to the rest of the drawing and thus remove overlaps. In very complicated drawings number labels will tend to be obscured by overlaps, in which case zooming is a valuable feature for detailed discernment. I here rely on hardware zooming as offered by many graphics terminals, such as the Plessey PT-100G.

DRAWSTRUCT creates a drawing as a Tektronix 4010/4014 compatible file or as a SUN pixrect file for viewing on the corresponding media as soft or hardcopy. As input it requires an ASCII file consisting of a list of the base-pairing which defines the structure. This file with the characteristic extension ".bpl", is supplied by the user or is created as output from the MONTECARLO or DYNPRO programs. The base-pairing list file is translated into a ".sstl" file which is an encoding of the base-pairing list in the form of hairpins. Commands for drawing the interrelated hairpins are then created to constitute the output ".pix" file.

STEMS

The finding of hairpins independently of a global structure is the purpose of this program. Normally, this is a simple task so long as no bulges or inner loops are allowed in the corresponding stems. But when this constraint is relaxed, the problem is then how to best choose among the competing possibilities. I have adopted two approaches to this problem: Monte Carlo and deterministic. Recalling that the stem of a hairpin may be considered to be a sequence of regions (a set of contiguous base pairs) separated by bulges or inner loops, both approaches first locate a potential hairpin by finding the closing base pair of its loop. This closing base pair is the first base pair of the first region. Having determined the extent of this first region, both

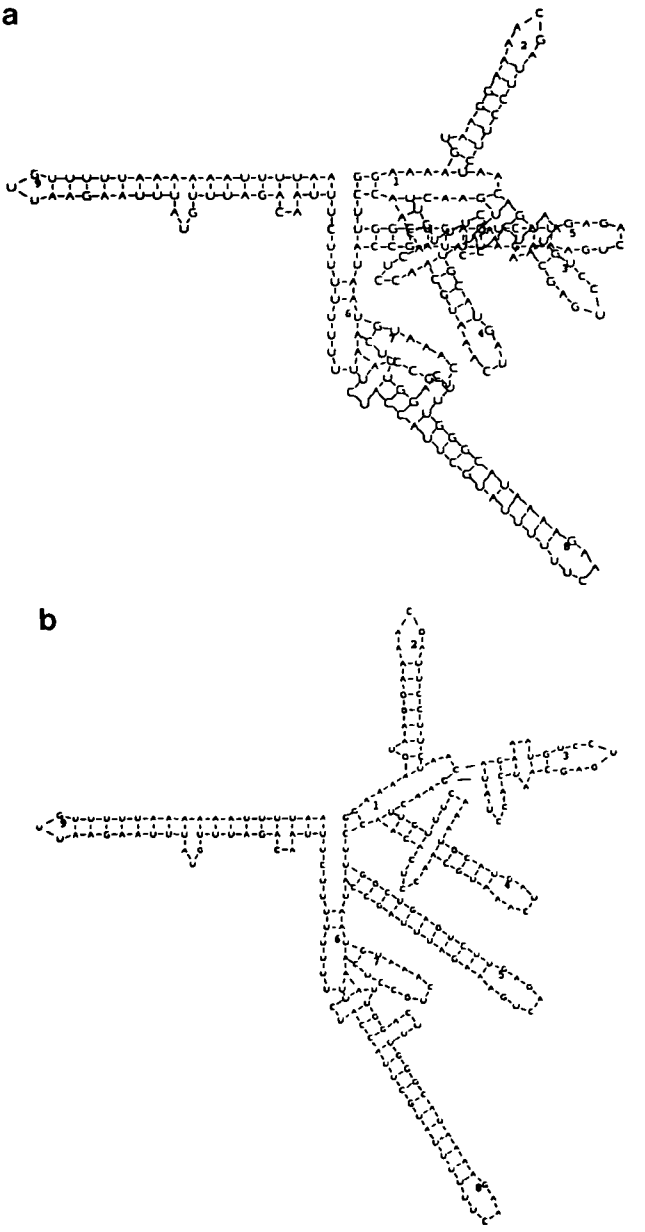

Fig. 1. a. RNA drawing with some overlaps. b. Same RNA with overlaps removed.

methods next determine all of the potential second regions which could follow the first one subject to the constraints on the intervening bulge or inner loop and such that their individual free energy G, consisting of base-pairing plus stacking energies plus destabilization due to the intervening bulge or inner loop, is negative. Given this population of potential second regions, the Monte Carlo method proceeds by selecting one of them with a probability proportional to its Boltzman exponential exp(-G/RT), while in the deterministic approach there is made a list of the X best ones for the construction of a hairpin tree with its root at the first region.

The Monte Carlo method of hairpin generation is naturally much faster than the deterministic one because no two of the hairpins it provides have the same first region. If the value of X for the deterministic approach were 1, then the number of hairpins found in both cases would be the same. A further limitation on the potentially explosive number of hairpins which could be obtained by the deterministic approach is provided by another parameter Y which limits the total number of hairpins to the Y best ones (in the sense of total free energy). The output of STEMS is exemplified by the following list of hairpins. Only the stems of same are shown, along with four numbers which identify the positions of each of the halves of the stem. The energy of the hairpins is also given, and the order of display corresponds to the most energetic ones being shown first. It will be noticed that the bases participating in pairing are capitalized.

```
** STEM (HAIRPIN) output for RNA sequence 'myc3' **

Sequence type: linear
Generation mode: random with seed 1
Temperature = 25.0 deg.C
Max hairpin loop allowed: 20
Max side of inner loop allowed: 6

Number of stems: 9
Best formation energy found: -12.62 Kcal
(-12.6)

98    UGGCUGAGUCUU    109
129   ACCGAUUUAGAA    118

(-10.2)

150   UGGacuuUGGGCAUAAAA    167
187   ACC::::AUUCGUAUUUU    174

( -6.0)
```

```
140   UGCC   143
162   ACGG   159

( -4.6)

60    UUGUuucaaaUGCAU    74
94    AACAcucca:ACGUA    81

( -4.5)

9     AAGGAA     14
25    UUCCUU     20

( -0.8)

1     GGAA      4
23    CCUU     20

( -0.4)

104   AGUC   107
115   UCAG   112

( -0.3)

164   AAAAGA   169
193   UUUUCU   188

( -0.3)

142   CC   143
152   GG   151
```

A stem list of this type can be used in a number of ways. Perhaps the most interesting one is for generating structure motifs of potential interest. Accordingly, there is provided the option of selecting from this list specified members for assembly into a base-pairing list which acts as the input to DRAWSTRUCT for rendering the motif.

MONTECARLO

The guiding consideration in this method of obtaining a global secondary structure is the generation of a population of structures on the basis of both energy and kinetic assumptions, as put forth in Martinez [1]. The idea is that stems with the best energy will tend to form first and that already-formed stems will tend to guide the formation of subsequent ones. A folding pathway is therefore hypothesized.

Significant changes have been made to the implementation previously reported on. Perhaps the most important of these is the option to use stems that can have bulges and inner loops. Generating such stems uses the corresponding Monte Carlo method outlined above in STEMS. Another important change is that the size of the stem competing populations is limited to a maximum number of best ones. When the initial number of best ones have been used up in forming a structure, the free

(unpaired) portions of the structure are scanned for potential stems and competition among these carried out to form detailed aspects of the structure. The folding is thus done in a recursive manner, with the folding at one level guiding the folding at the next level of refinement.

A user-adjustable parameter Max_Folds specifies the number of independent foldings to be done. Of these, the program saves only the distinct ones and reports their energy and frequency of occurrence. A composite file is created containing the base pair lists of each of the distinct structures, and this composite file is used by a display routine to interactively create individual ".bpl" files for input to the DRAWSTRUCT program. Thus, as individual structures of the generated population are viewed, they can be interactively modified for removal of any overlaps. Structures may also be viewed in text-type format exemplified by the following list.

MONTECARLO folding of 'myc3'

Constraints: none

Parameters:

NFoldings = 10
Max-Stems-Factor = 5
SEed = 1
Max-Bulge-Size = 6
Loop-Destab = 1
Max-Hairpin-Size = 240

struct #1: frequency = 1/10, energy = -32.64 Kcal

```
stem 1: bifurcates into stems 2 3 4

3     AAAAg::::UAaggAAAACGAuuccUUCUaac(2) (3)u(4)ua
241   UUUUaaaaaAU:::UUUUGUU::::AAGA

stem 2: a hairpin

31    AGaaauGUccUGA::::GCA::::AUCaccuAUGaacUUGuu
97    UC::::CAacACUccaaCGUaaacUAG::::UACguaAACu


stem 3: a hairpin

98    UGGCUGAGUCUUgaga
129   ACCGAUUUAGAAaguc


stem 4: a hairpin

131   AAUGUAAA:CUGccuc::AAAu::::::::UGGacuuUGGGCAUAAAAgaa
217   UUAUGUUUaGACaauuucUUUuuuuuuucuACC::::AUUCGUAUUUUuuc
```

When comparing results of this probabilistic method of folding with those of the dynamic programming approach to be described below, I have found it instructive to consider relaxing the destabilizing effect of hairpin loops on the grounds that there may indeed be base stacking in the single stranded portions and hence that hairpin loops are not quite as random as is usually supposed. This has enhanced the production of characteristic cloverleaf patterns for tRNA. I have therefore included this feature as an option pending the availability of data which will give specific information regarding the influence of base stacking in single strands.

DYNPRO

This program is our implementation of the Zuker and Stiegler [3] dynamic programming algorithm. It produces two kinds of output. One is a ".bpl" file for input to DRAWSTRUCT in case a drawing is desired, and the other is a text listing of the structure in a form which shows the hairpins and the relations between them, as illustrated above for the MONTECARLO text output. The structure shown below was determined under the same conditions as the one obtained by the MONTECARLO method. The energy is significantly more for this particular case (-57 vs -33 kcal) even though there is some base-pairing in common. This is what generally happens. Unless the structure is a very sharp one in the sense that there are practically no competing structures of about the same energy but different motifs, the two methods will give divergent results. This is because the MONTECARLO approach is biased along hypothesized folding pathways corresponding to strong local structures. Nevertheless, it is instructive to quickly generate global structures corresponding to local energy minima because a population of these will generally possess some common stems and it is these which should be expected to participate in structures of absolute minimum energy as found by the dynamic programming methodology. Given that it is a difficult matter to generate significantly different, competing structures with the dynamic programming method, the utility of the Monte Carlo approach becomes more evident.

```
DYNPRO structure of 'myc3'

Energy = -57.00 Kcal

stem 1: bifurcates into stems 2 3

4      AA(2)(3)
241    UU


stem 2: a hairpin

6      AGuAAGGAAaac
27     UC:UUCCUUag


stem 3: bifurcates into stems 4 9

28     AAc(4)u(9)
239    UU
```

```
stem 4: bifurcates into stems 5 8

31    AGA(5)(8)
190   UCU

stem 5: bifurcates into stems 6 7

34    AAuGUCC::::UGAG:CAa::UcACc:UAugaac(6)ccu(7)
156   UU:CAGGuuaaACUCcGUcaaA:UGuaAU

stem 6: a hairpin

60    UUGUuucaaaUGCAUgau
94    AACAcucca:ACGUAaac

stem 7: a hairpin

98    UGGCUGAGUCUUgaga
129   ACCGAUUUAGAAaguc

stem 8: a hairpin

157   UGG:::GCAUAAAAgaa
187   ACCauuCGUAUUUUuuc

stem 9: a hairpin

192   UUUUUU:::::::UUCUUu:A:ACaga
237   AAAAAAuuuuuguuAAGAAuuUaUGuuu
```

CONSTRAINTS

This is a command level used for entering, modifying or looking at previously specified base pairing constraints. A base pairing constraint is specified in the format 'basepos#1 [basepos#2] code'. The basepos#2 entry is optional. A code value of 0 means 'do not base pair' and a 1 means base pair with high priority. If two base numbers are given, the code refers to the pairing of the first with the second; otherwise it refers to the given base with respect to the rest of the sequence. As in the Zuker and Stiegler program, "bonus energies" are used to force desired base-pairs.

Example 1: 23 45 0

This means that the bases at positions 23 and 45 are not to base pair with each other.

Example 2: 23 1

This means that the pairing of the base at position 23 with some other base is of high priority.

ENERGY RULES

The energy rules used by STEMS, MONTECARLO and DYNPRO are those of Salzer [4]. The program DRAWSTRUCT also uses these rules to calculate the energy of a given structure which has been defined by a list of base pairs. Future enhancements are intended to provide for user entry of other rules that will include single strand, base-stacking stabilization and a choice among various possibilities for dealing with the desabilization which can occur in bifurcations. The rule I have currently adopted for this kind of destabilization corresponds to that of an inner loop closed by the base pairs AU and CG.

CONTEMPLATED ENHANCEMENTS

In addition to some flexibility in the choice of energy rules for orthodox-type structures (no knots), consideration is also being given to the problem of "pseudo-knots" whereby adjacent hairpin loops are allowed to base pair in a manner which effectively increases the length of one of the stems . Current indications are that such an extension is best handled via the Monte Carlo method of folding. In this method the addition of a stem to the current structure depends on the amount by which it reduces the free energy of same. The calculation of this reduction can include a term corresponding to the potential formation of pseudo-knots with the loops of already formed stems. Also to be considered is the problem of how to draw the pseudo-knots. It is not clear how this can be easily done in two dimensions.

Some time ago I constructed a folding program based on the algorithm of Studnicka, et al [5] whereby a list of perfect stems (no bulges or inner loops) was first generated and then from these there was extracted an optimal orthodox structure. The drawback of this approach, while giving good results with regard to the prediction of tRNA structures, is the lack of detail regarding bulges and inner loops and the fact that in contrast to full dynamic programming (resolution down to the single base pair) which are of cubic time order in sequence length, it was of quintic time order and therefore only useful for small sequences. At that time I had not developed the concept of successive refinement as used in the current MONTECARLO method nor of generalized stems which can include bulges and inner loops. A revival of the deterministic regions method is therefore contemplated which will take advantage of these innovations in order to substantially reduce computation time and obtain increased resolution.

A practical deterministic regions method will provide still another means of generating structures, though from a biased point of view in the sense of implying a folding pathway. It will be more like the pure dynamic programming approach in the sense of giving better global-type of results but still retain the kinetic (folding pathway) feature. To my way of thinking, it will correspond, functionally, to the folding method proposed by Dumas and Ninio [6]. In addition, the pseudo-knot feature can also be attained because of the manner in which optimal structures are extracted from stem lists.

PROGRAM DESCRIPTION

RNAFOLD is written in the C language, it provides menu driven interaction with the user for calling on the various programs and adjusting parameters, is completely self-documented, has a resident size of 163 Kbytes and consists of 29 modules.

The modular aspect permits the convenient addition of new command levels as required for accommodating new folding schemes. The changing of parameters values or base-pairing constraints is enhanced by the feature of being able to save changes and reloading them for future runs.

REFERENCES

1. Martinez,H.M.(1984) Nucl.Acids Res.12:323-334.
2. Shapiro,B.A.,Maizel,J.,Lipkin,L.E.,Currey,K.,Whitney,C. (1984) Nucl.Acids Res.12:75-88.
3. Zuker,M. and Stiegler,P, (1981) Nucl. Acids Res.9:133-148.
4. Salser,W.(1977) Cold Spring Harbor Symp. Quant. Biol. 42:985-1002.
5. Studnicka,G.M.,Rahn,G.M.,Cummings,I.W.,Salser,W.A.(1978) Nucl. Acids
6. Dumas,J.P. and Ninio,J. (1982) Nucl. Acids Res.10:197-206. Res.5:3265-3387.