Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Journal of Molecular Graphics and Modelling 25 (2006) 514–531

Journal of Molecular Graphics and Modelling

www.elsevier.com/locate/JMGM

Graphical exploratory data analysis of RNA secondary structure dynamics predicted by the massively parallel genetic algorithm

Bruce A. Shapiro [a,*], Wojciech Kasprzak [b], Calvin Grunewald [a], Javed Aman [a]

[a] *Center for Cancer Research Nanobiology Program, National Cancer Institute, Building 469, Room 150, Frederick, MD 21702, United States*

[b] *Basic Research Program, SAIC Frederick, NCI-Frederick, Building 469, Room 150, Frederick, MD 21702, United States*

Received 13 December 2005; received in revised form 5 April 2006; accepted 6 April 2006
Available online 28 April 2006

Abstract

Studies indicate that RNA may enter intermediate and multiple conformational states, which may impact gene expression and molecular function. It is known that the biologically functional states of RNA molecules may not correspond to their minimum energy conformations, that kinetic barriers may trap the molecule in a local minimum, that folding often occurs during transcription, and that cases exist in which a molecule will transition between one or more functional conformations. Thus, methods for simulating the folding pathway and dynamic behavior of an RNA molecule are important for the prediction of RNA structure and its associated functions.

We have developed several data mining techniques guided by interactive visualization tools associated with our massively parallel genetic algorithm for RNA/DNA secondary structure prediction, MPGAfold, and StructureLab analysis workbench. Most of the methods and tools are also applicable to dynamic programming algorithm (DPA) folding data analysis. When applied to MPGAfold results these methodologies are used to determine the significant intermediate and final structures associated with co-transcriptional and full length RNA folding. Since the genetic algorithm is essentially stochastic, multiple runs are required to develop a consensus understanding of an RNA structure. The interactive visualizations facilitate interpretation of results from sequential or full length individual MPGAfold runs, final results of multiple folding runs, including multiple population sizes, and the results from multiple RNA sequences of one family. This paper describes several of these techniques and shows how they are used to help solve this highly combinatoric problem.

Keywords: RNA secondary structure; HIV-1 structural metastability; Folding pathway; Visual datamining; Massively parallel genetic algorithm (MPGAfold); StructureLab analysis workbench

1. Introduction

The bioinformatics revolution has led to a tremendous increase in the availability of data on gene location, expression, and function for thousands of species. Because of this vast quantity of data, time and resources are often lacking for in depth experimental analysis of genes and gene products. In the past, proteins were the main focus of attention for detailed structural analysis. However, more recently RNA structural studies have become very important to the understanding of complex biological systems.

The number of ways that RNA can interact with its environment is extensive. Structure and structural transitions are important in post-transcriptional regulation of gene expression [1], such as with RNAi [2] and riboswitches [3], intermolecular interactions [4], splice site recognition [5], and ribosomal frameshifting [6], to name a few contexts. Ribozymes, for example, constitute a class of RNA molecules whose sequence exists primarily to define their enzymatic properties [7]. The RNA folding problem, i.e. the determination of RNA secondary structure and ultimately three-dimensional structure and function, is a significant area for the use of computational approaches. As with most such applications, the problem of RNA structure determination is a difficult one. The number of possible secondary structures given a particular sequence varies on the order of 1.8^n for a sequence of n nucleotides [8]. Approaches to the problem are numerous and varied. A wide range of biochemical and biophysical methods may be used to examine RNA secondary and tertiary structure. These methods generally search experimentally for an effect on a sequence/structure of a

[*] Corresponding author. Tel.: +1 301 846 5536; fax: +1 301 846 5598.
E-mail address: bshapiro@ncifcrf.gov (B.A. Shapiro).

1093-3263/$ – see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jmgm.2006.04.004

molecule by probing it for accessibility [9,10], calculating optical absorbency [11], or by measuring its electrophoretic migration rate over a temperature gradient [12]. A given structure generally is verified through sequence comparison analysis, searching among members of a family for compensatory base changes that would maintain base pairedness in equivalent regions [13–18]. In addition, the three-dimensional structure of these molecules may be elucidated by X-ray crystallography or nuclear magnetic resonance techniques [7].

All of this relatively direct data often is supported, or at times even replaced, by theoretical structure calculations. The most familiar variety are those that are derived from dynamic programming algorithms (DPA) such as Mfold [19,20], RNAstructure [20,21], or RNAfold [22], which search for a molecule's thermodynamically optimal structure, as well as a series of suboptimal structures. When the object is a secondary structure, which can be defined as a list of base paired and single-stranded regions (stems and loops), thermodynamic calculations are straightforward. Stems tend to stabilize a structure and most loops tend to destabilize it, and the energies of these stems and loops are additive. Thus, a search for biologically relevant structures is driven by the assumption that a molecule will tend to fold spontaneously into structures that minimize its global Gibbs free energy with respect to the unstructured state. Dynamic programming methods can also incorporate several variously limited classes of pseudoknots at the cost of increasing the complexity; $O(n^6)$ for Pseudoknots of Rivas and Eddy [23], $O(n^5)$ for NUPACK [24], and $O(n^4)$ for pknotsRG [25]. Without considering pseudoknots the dynamic programming algorithms can run in $O(n^3)$ time [20]. Recently two heuristic algorithms capable of computing structures with pseudoknots were published; ILM [18], and HotKnots [26], which are less complex and can deal with longer sequences but do not guarantee the minimal energy solutions.

Searching experimentally and theoretically for these optimal or suboptimal structures, however, is often insufficient. The biologically functional state of a given molecule may not be the optimal state, as it was illustrated in [1], for example. The issue then is how does one determine the relevant suboptimal structures? A structured RNA molecule, moreover, is not a static object. A molecule may pass through several active and inactive states over its lifetime, due to the kinetics of folding, to the simultaneity of folding with transcription, i.e. sequence elongation, or to interactions with its environment. A molecule may become trapped in a local energy minimum with a high-energy barrier to surmount before reaching a more stable state. Programs such as RNAshapes [27–29] and Sfold [30,31] produce a relatively small set of representative structures for the usually very large solution spaces produced by the dynamic programming algorithm, thus simplifying the search for relevant solutions. However, based on our experience, they may miss some conformations or assign extremely low probabilities to representative shapes known to be biologically significant from experimental data. The KINEfold webserver [32,33] simulates folding kinetics, but it is still relatively limited by speed and sequence size.

Thus, RNA structure analysis has not reached the level where the user could submit a sequence and obtain "the answer" after one mouse click. There may not even be one answer, as the real structure may have "breathing" regions or oscillate between conformations. Interactive analysis of results produced by multiple alternative structure prediction programs is still necessary and our software workbench described below has been developed to provide a set of tools to achieve such analysis flexibility. We have developed methods using a massively parallel genetic algorithm, MPGAfold, that have proven highly amenable to the exploration of RNA secondary structure folding pathways [1,34–39] MPGAfold was designed using the same basic energy considerations as the dynamic programming algorithms; that is, with thermodynamic calculations to optimize the global free energy of an RNA molecule, including the coaxial stacking energy calculations for multi-branch loops (EFN2) [20] performed at every generation. The EFN2 calculations were also recently added to the run-time processing in the DPA implementation, RNAstructure 4.2 [21]. The genetic algorithm is reasonably successful at finding optimal or near-optimal equilibrium structures, including H-type pseudoknots, given a particular sequence. The properties of this massively parallel, iterated, stochastic algorithm, however, have also been shown to be well suited to the problem of predicting the dynamic folding process of a given molecule by focusing on significant intermediate and final structures. In addition, the algorithm allows for the incorporation of some types of experimental data, allowing it both to verify and to predict the outcome of experiments under known conditions. MPGAfold operates on a population of thousands of possible solution structures, evolving them toward thermodynamic fitness. It may be run multiple times and in multiple phases. StructureLab, an interactive RNA structure analysis workbench [40,41], has proven indispensable in analyzing the large quantities of data generated by the genetic algorithm. StructureLab, in combination with a set of MPGAfold visualizer maps is used to mine the genetic algorithm's output for useful data. The visualization and analysis tools we are going to discuss present views of the generated data from several partly overlapping, but unique perspectives. Ultimately, these perspectives are combined to obtain a fuller understanding of the folding patterns of the RNA in question. The tools are interactive and are meant to be used in a process of iterative refinements of the interpretation of the data.

All the visual data mining examples depicted in this paper are derived from our two recent studies; one on the folding characteristics of the HIV-1 leader sequence [42,43] and the other on the differential folding of the HIV-1 5′ and 3′ poly(A) signals [44]. In vitro studies of the Human Immunodeficiency Virus Type 1 (HIV-1) 5′ untranslated leader region have demonstrated that it can fold into two alternative functional structures [4,45–48]. One conformation, called Branched Multiple Hairpins (BMH) or simply branched, contains two key stem-loop motifs, the SL1 and the poly(A) SL. SL1 contains the Dimer Initiation Site sequence (DIS) with its self-complementary heaxamer motif exposed in SL1's hairpin loop, crucial in the dimerization and packaging of HIV-1.

The poly(A) SL occludes the 5′ polyadenylation signal. The LDIs or linear conformations rearrange these two BMH motifs into long distance interactions forming extended ‘‘linear’’ motifs, which occlude the DIS and expose the poly(A) signal. The BMH conformation is proposed to be dimerization and encapsidation competent, while the LDI conformations are thought to be translation competent (in [4] and references therein). These two conformations and ways of elucidating them will be illustrated throughout the text.

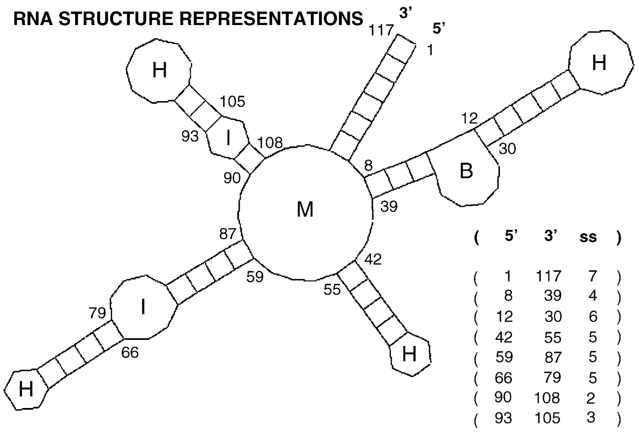
2. Methods and tools

2.1. RNA secondary structure representations

An RNA sequence can be viewed as a string derived from an alphabet consisting of the letters (A, G, C, U). These letters represent, respectively, the nucleic acid bases, adenine, guanine, cytosine and uracil. The bases tend to form Watson–Crick base pairs, i.e. G–C and A–U and a wobble base pair G–U that are stabilized by hydrogen bonds and base stacking interactions. Other base pairs are possible, but for the sake of brevity will not be brought into the current discussion.

The interaction of these bases will form different types of motifs, which are depicted in Fig. 1. Each crosshatched line represents a base pair, while other positions represent bases that are not normally paired and thus form loop regions. Loop regions are labeled according to their motif type. M, I, B, and H stand for multibranch loop, internal loop, bulge loop and hairpin loop, respectively. The region table, which represents the topology of the structure, is also shown. The region table represents individual helical stems sorted based on their 5′ positions. Thus, the first stem, represented by the triple (1 117 7) is a stem whose 5′ position is 1, its 3′ position is 117 and its size is 7.

Each loop motif and hydrogen bonded stem contributes to the free energy of the secondary structure (the more negative the free energy the more stable the structure). For the most part, loop regions tend to destabilize the structure (more positive free energy), while base paired regions tend to stabilize the structure (more negative free energy). Energy rules, with different degrees of context sensitivity, are used to compute the free energy of a structure.



(	5'	3'	ss	)
(	1	117	7	)
(	8	39	4	)
(	12	30	6	)
(	42	55	5	)
(	59	87	5	)
(	66	79	5	)
(	90	108	2	)
(	93	105	3	)

Fig. 1. A typical RNA secondary structure indicating the various morphologies that are commonly present. B, bulge loop; H, hairpin loop; I, internal loop and M, multibranch loop. Also illustrated is the stem table that defines the topology of the given structure.

2.2. The massively parallel genetic algorithm

A massively parallel genetic algorithm MPGAfold was developed to predict RNA secondary structures from a given sequence. It has been described in detail elsewhere [1,34–39], but for completeness it will be briefly introduced below. The genetic algorithm borrows from the biological concepts of evolution and the survival of the fittest [49–51]. In its current implementation, the algorithm is capable of running on several different computer architectures [38], and most recently, it has been ported to LINUX clusters using MPI-2.

The algorithm uses a mesh type representation for the population, where a population element is defined to be a ‘‘maturing’’ RNA secondary structure. Eight neighbors surround each population element (north, south, east, west, northeast, northwest, southeast and southwest (Fig. 2)). Two parents are chosen from the nine possibilities based upon a biased free energy selection criterion. Two children are created from these two parents by randomly mutating in stems from the stem pool (the set of all possible fully complementary stems that can be generated from the given sequence) and by recombining stems from the two parent structures, with possible conflict-driven stem peelback [38,39]. The child with the best fitness (lowest free energy) is chosen to replace the population element in the center of the eight-neighbor region. This operation can be thought of as occurring in parallel on all

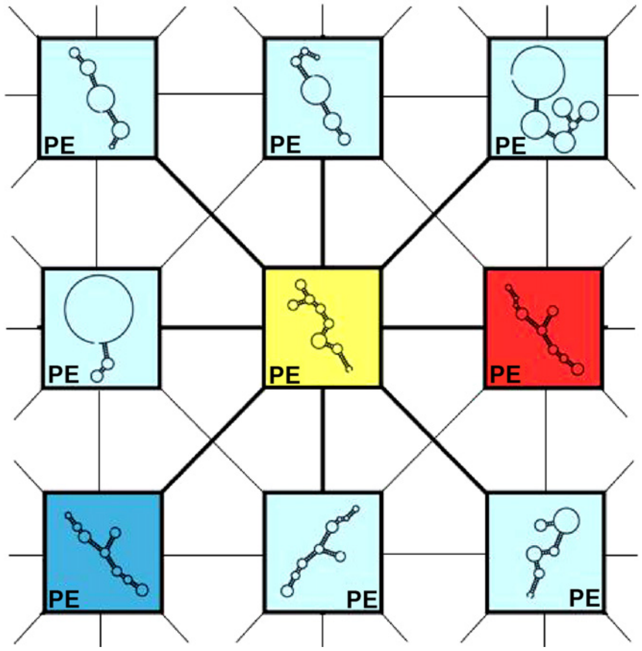


Fig. 2. Representation of the population element layout and connectivity showing a small window, which, for example, might represent a portion of a 16 K population in MPGAfold. The red and dark blue boxes represent chosen parents and the yellow box represents the placement for a new structure in a 3 × 3 neighborhood.

possible 3 × 3 eight neighbor toroidally wrapped regions, thus producing, with a 16 K population, 16 K new population elements at each generation. The algorithm iterates over several generations until a convergence criterion is satisfied which measures the relative stability of the population as a whole. Stability of the population is induced by an annealing mutation operator, which gradually reduces the number of mutations of ensuing generations [34].

The genetic algorithm is normally run with several different population configurations each varying by a power of two. Thus, typical runs might involve 2 K through 128 K and sometimes 256 K populations. Each population size is normally run 20 times to develop a consensus. A given population size can in turn be run with a power of two number of physical processors. The algorithm scales almost linearly with the number of physical processors. Thus, for example, doubling the number of physical processors will improve the speed by about a factor of 2.

Population size variation has the interesting property of capturing RNA secondary structures that are representative of significant intermediates. That is, as an RNA sequence is folding, it will sometimes form intermediate conformations that are themselves functional or are important for the folding pathway of the RNA for reaching its final state. Typically, at lower population levels, the algorithm will converge to less fit (higher free energy) solutions, which are indicative of these significant intermediates. At higher population levels, the evolving structures will pass through these intermediates on their way to possibly more fit significant intermediates or their low free energy final states, which do not have to correspond to the minimum free energy structure produced by the dynamic programming algorithms.

Many other features exist within the algorithm including co-transcriptional folding (also known as sequential folding or folding with sequence elongation), biased use of certain motifs, the use of various sets of energy parameters, run-time determination of EFN2 coaxial stacking calculations, Boltzmann filter, conflict driven peelback, multiphase runs and ‘‘sticky stems’’. The latter is a way of including known biological information. In addition, H-type pseudoknots, the interactions of a hairpin loops and free base regions, can be calculated. A description of some of these features can be found in [1,36–39,41]. An important issue and the main focus of this paper is the analysis of the significant amounts of data that are produced by the algorithm, most of which is important for forming an understanding of the folding characteristics of the given sequence. In our earlier publications [1,42–44] we have shown that the intermediate folding states captured by MPGAfold correspond to the states of the folding pathways in several systems. Both visual and statistical means are used to approach this problem, most of which StructureLab, our RNA/DNA structure analysis workbench handles.

2.3. Visualization of MPGAfold run-time population dynamics

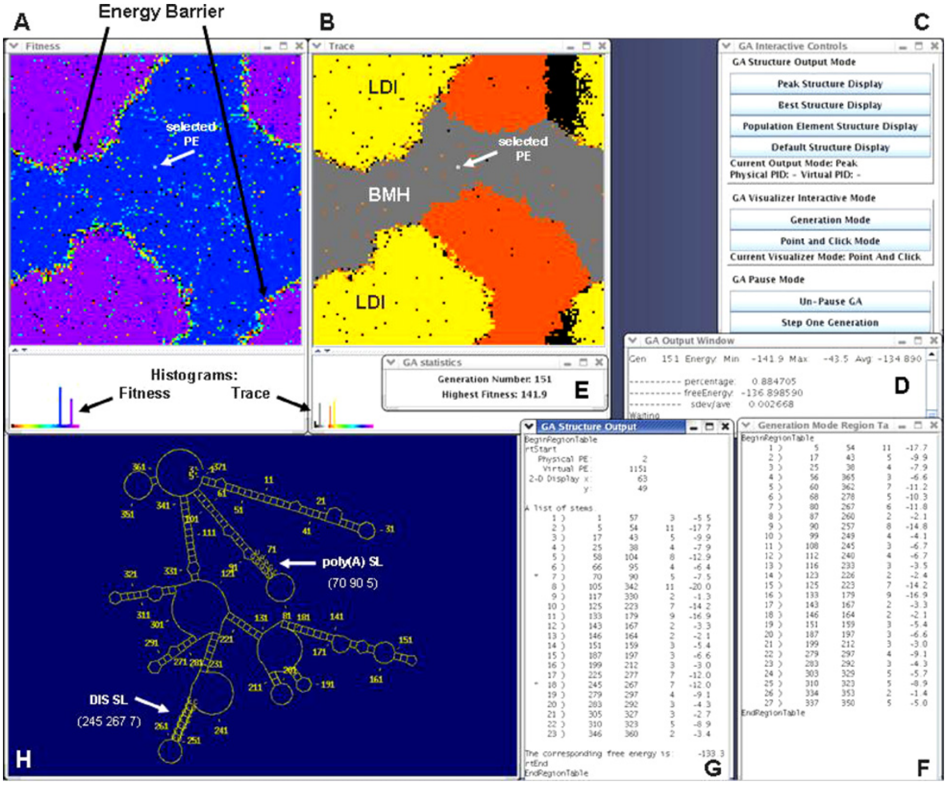
Three different $n \times m$ two-dimensional dynamic graphical population maps can be created at each generation (or chosen increment) and viewed as the genetic algorithm is running. The values of n and m are determined by the power-of-two population size. Thus, a 128 × 128 square region would represent a 16 K population, while a 64 × 128 rectangular region would represent an 8 K population, etc. Each pixel in a map represents the contents of a particular population element. By pointing at a particular pixel in one of the maps, a region table and a drawing (see Section 2.6) representing the corresponding structure will be displayed, with all the user-selected secondary structure drawing preferences, such as annotations and labels.

2.3.1. Fitness map

In the fitness map, each pixel is color-coded based on the fitness of the structure evolving in the corresponding element. Red, at one end of the color spectrum indicates poor fitness (high free energy) while purple, at another end of the color spectrum, indicates good fitness (low free energy). Interesting characteristics of the folding landscape are sometimes discernable from the fitness map. Sometimes population clusters, represented by two different uniform colors (and therefore different fitness values) may be surrounded by a lower fitness region (Fig. 3A). This is representative of an ‘‘energy barrier’’, i.e. the RNA has to unfold somewhat to transition from the higher free energy state to the lower free energy state. Other times, the transition is smoother without the existence of the intermediate barrier, indicating that relatively minor transitions are probably taking place. We can identify the differences between the structures from regions of different fitness by interactively extracting the corresponding regions' information. We can utilize the trace map information to detect the presence of key motifs in the fitness map (see next section). We can also draw the secondary structures of interest, when MPGAfold is run in synchronization with StructureLab (see Section 2.6). For example, in the snapshot of the HIV-1 MN folding run presented in Fig. 3A, the inner region of blue pixels corresponds to a population of the lower fit BMH (branched) conformers, while the region of purple pixels corresponds to the better fit LDI (linear) structures.

2.3.2. Trace map

The trace map, which has a one-to-one positional correspondence with the fitness map, allows one to follow the formation and disappearance of helical stems that one suspects or knows exist *a priori*. Each individual occurrence of a stem in a structure is color-coded. If more than one stem is present from the trace list, then the pixel is shaded gray. Its gray value is determined by the percentage of stems found in a structure from the list. If all the stems from the list appear, then the pixel is coded white. Thus, the trace map allows one to understand the dynamic formation of individual stems as the algorithm is running and to determine visually when many population elements acquire the depicted stems. Fig. 3B illustrates the trace map that is following the propagation of stems for known motifs in a given sequence. The illustrated trace map is derived from the same generation as the fitness map in Fig. 3A. By viewing the fitness map together with the



Current Output Mode: Peak
Physical PID: - Virtual PID: -

Current Visualizer Mode: Point And Click

Generation Number: 151
Highest Fitness: 141.9

```
Gen    151 Energy  Min   -141.9  Max    -43.5  Avg  -134.890

---------- percentage:    0.884705
---------- freeEnergy:  -136.898590
----------   sdev/ave:    0.002668
Waiting
```

```
BeginRegionTable
rtStart
   Physical PE:           2
    Virtual PE:        1151
 2-D Display x:          63
             y:          49

A list of stems
      1 )      1     57      3    -5.5
      2 )      5     54     11   -17.7
      3 )     17     43      5    -9.9
      4 )     25     38      4    -7.9
      5 )     58    104      8   -12.9
      6 )     66     95      4    -6.4
  *   7 )     70     90      5    -7.5
      8 )    105    342     11   -20.0
      9 )    117    330      2    -1.3
     10 )    125    223      7   -14.2
     11 )    133    179      9   -16.9
     12 )    143    167      2    -3.3
     13 )    146    164      2    -2.1
     14 )    151    159      3    -5.4
     15 )    187    197      3    -6.6
     16 )    199    212      3    -3.0
     17 )    225    277      7   -12.0
  *  18 )    245    267      7   -12.0
     19 )    279    297      4    -9.1
     20 )    283    292      3    -4.3
     21 )    305    327      3    -2.7
     22 )    310    323      5    -8.9
     23 )    346    360      2    -3.4

The corresponding free energy is:        -133.3
rtEnd
EndRegionTable
```

```
BeginRegionTable
      1 )      5     54     11   -17.7
      2 )     17     43      5    -9.9
      3 )     25     38      4    -7.9
      4 )     56    365      3    -6.6
      5 )     60    362      7   -11.2
      6 )     68    278      5   -10.3
      7 )     80    267      6   -11.8
      8 )     87    260      2    -2.1
      9 )     90    257      8   -14.8
     10 )     99    249      4    -4.1
     11 )    108    245      3    -6.7
     12 )    112    240      4    -6.7
     13 )    116    233      3    -3.5
     14 )    123    226      2    -2.4
     15 )    125    223      7   -14.2
     16 )    133    179      9   -16.9
     17 )    143    167      2    -3.3
     18 )    146    164      2    -2.1
     19 )    151    159      3    -5.4
     20 )    187    197      3    -6.6
     21 )    199    212      3    -3.0
     22 )    279    297      4    -9.1
     23 )    283    292      3    -4.3
     24 )    303    329      5    -5.7
     25 )    310    323      5    -8.9
     26 )    334    353      2    -1.4
     27 )    337    350      5    -5.0
EndRegionTable
```

Fig. 3. MPGAfold visualizer and StructureLab interface windows illustrating the HIV-1 5′ UTR structure example discussed in the text. Windows shown in A and B contain structure fitness and region trace maps of a 16 K population run. The pixel positions in each 128 × 128 map correspond to each other. (A) Fitness map representing the later stages of an MPGAfold run. Each pixel of this map is color-coded based upon the fitness value of the structure in the population element. The rugged region annotated as "energy barrier" indicates that the structure in the areas on each side of the barrier are quite different and that the barrier represents structures that are undergoing unfolding to allow for a transition. (B) Trace map allows one to follow the occurrence or disappearance of individual user-specified stems in the population. The stems are each color-coded when they appear in isolation in a structure. In the trace shown, the yellow areas correspond to structures containing the key LDI (linear) stem (see Fig. 7B). Red pixels depict the population elements containing the poly(A) SL. Their presence indicates evolving structures with one of the two key elements of the BMH (branched) conformation. If several of the traced stems (but not all) appear in a structure, a gray value is used to represent the number of traced stems that occur together. In this case gray pixels denote structures with two BMH stems; the poly(A) SL and either of the two SL1 variants (see Figs. 5 and 7B). White pixels depict the occurrence of all followed stems in the population element (not in the example shown). The population fitness and trace histograms are shown at the bottom of the Fitness Map and the Trace Map windows. Both histograms show two peaks corresponding to the two dominant conformations, the LDI and the BMH. Their color-codes are the same as those used in the respective Map windows. (C) MPGAfold visualizer's control window. (D and E) Population statistics display windows. (F) Region table representing a population histogram peak structure of the LDI-type (yellow areas in the Trace Map and the yellow peak in the Trace Histogram). (G) Region table interactively extracted from the selected PE (indicated with white arrows in A and B). Traced stems present in this table are marked with asterisks. (H) RNA Secondary structure automatically drawn by StructureLab based on the interactively selected PE's region table (shown in G). This drawing illustrates a maturing BMH-type HIV-1 MN structure, with the two key traced BMH stems labeled with their respective sequences and identified with white arrows and name labels. Drawing features (untangling, position annotations and traced stem sequence labeling) are set in StructureLab.

trace map we can identify the known (traced) conformation types and their relative fitness. In the example shown in Fig. 3B, the key stems or motifs indicate the presence of the linear-type structures (yellow pixels, labeled LDI) and the branched-type conformers (gray pixels, labeled BMH) in the folded HIV-1 MN sequence. Red pixels correspond to structures containing only one of the BMH key stems, the poly(A) SL. In addition, a region table from the selected PE (shown in Fig. 3A and B) can be displayed, as shown in Fig. 3G. In it, the traced stems are marked with asterisks. Thus, using these two MPGAfold maps, we can visualize the process of maturation (increasing fitness) and associate it with specific conformations. In addition we can determine the population distributions of the conformations.

2.3.3. Pseudoknot map

Similar in concept to the other maps, this map follows the occurrence of pseudoknots in individual structures (not shown). The color-coding used in this case indicates the number of pseudoknots that are predicted in any given structure. Because of the pixel correspondences that exist between the maps, one could, for example, follow the formation of a previously known or expected pseudoknot by visually inspecting the pseudoknot map and the trace map together. One can use the mouse pointer to point at a pixel in any of the maps to extract the corresponding underlying data.

2.3.4. Population histogram

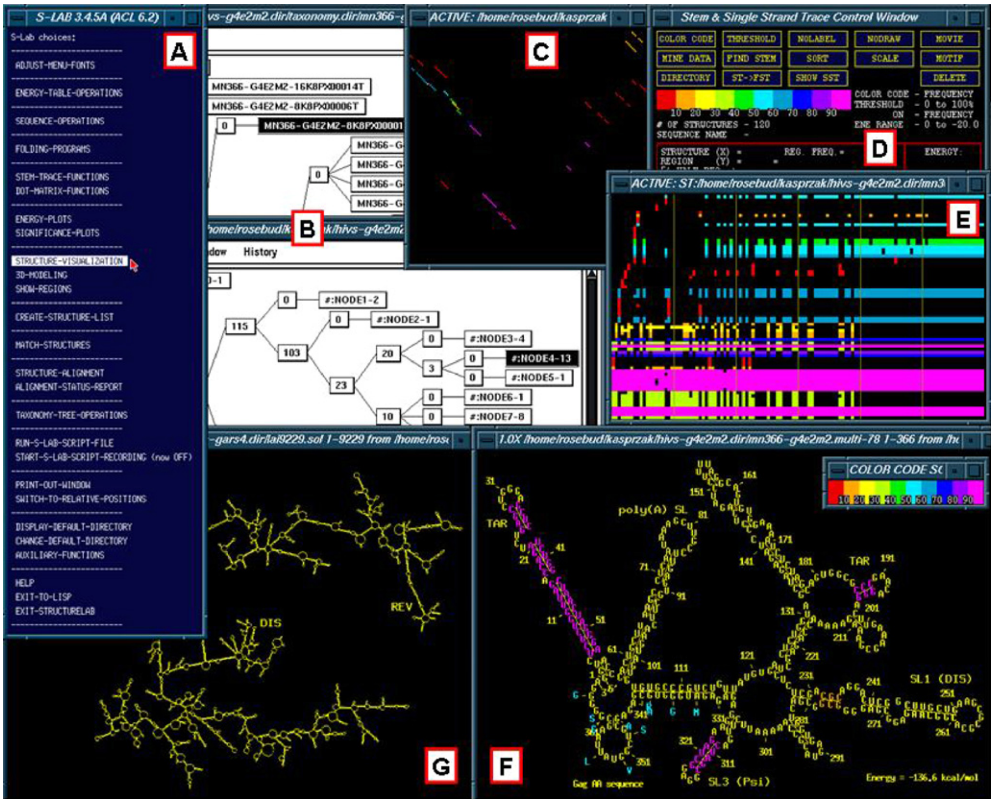
While the genetic algorithm is running the display of a population histogram can be activated, which for each generation (or chosen increment) will depict a color-coded histogram representing the fitness, trace or pseudoknot distributions for the entire population. The fitness and trace histogram distributions are visible in Fig. 3A and B, respectively. When the genetic algorithm starts most of the population, as expected, is distributed in the poorer fitness region. As the algorithm proceeds one can see the gradual redistribution in the histogram, showing more and more of the population having improving fitness.

At times, the population histogram will indicate the presence of multiple significant peaks. This has been correlated with the existence of competing populations of structures. In the case of the HIV-1 folding shown here, MPGAfold predicted two metastable conformational states with similar free energies, the BMH (branched) and the LDI (linear) [4,42,43,45–48]. The population histogram associated with the fitness map, visible at the bottom of Fig. 3A, shows two peaks corresponding to these two dominant conformations. The color-code used here reflects the fitness of the conformers. The population histogram associated with the trace map (Fig. 3B) shows the distribution of the structural motifs associated with the (yellow) LDI and BMH (gray) conformers. The color-code of the trace histogram was explained in the Trace Map section and is independent of that used in the fitness histogram.

2.4. Data generated by the genetic algorithm

The genetic algorithm is capable of generating several different types of data files, which can later be used for analyzing results. Setting various parameters in the input to the genetic algorithm can control the type and number of files generated. The types of files are enumerated below:

(1) Solution files—As MPGAfold is running, for each generation (or specified stride) a file is generated which represents the best current structure in the population or the current consensus structure for the generation. Considering that there may be as many as 700–800 generations per run (depending on the sequence size) there may be as many as 16,000 files generated in 20 runs, if one captures all the structures for each generation. To reduce the amount of data output, it is also possible to generate results only at the end of a run. In this case only the final best or consensus structure will be produced.
(2) Pseudoknot files—These files are associated with each solution file (see 1 above). They indicate those stems that will form pseudoknots.
(3) EFN2 files—These files are associated with each solution file (see 1 above). They indicate the coaxial stacking present (if any) between stems in multibranch and other loop constructs. These files are generated only if the EFN2 energy rule set is used for folding.
(4) Solution files representing the top "*n*" structures in the population—This permits the evaluation of several terminating structures at the end of a run.
(5) Histogram file representing the energy distribution in the population—This file will contain the population counts for each energy level found in a given population. This is a text version of the data displayed in the population histogram mentioned above. It is particularly useful for determining



COLOR CODE - FREQUENCY
THRESHOLD - 0 to 100%
ON - FREQUENCY
ENE RANGE - 0 to -20.0
OF STRUCTURES - 120
SEQUENCE NAME -
STRUCTURE (X) = REG. FREQ. =
REGION (Y) =
ENERGY:

Fig. 4. Depiction of some of the capabilities of StructureLab. Shown in a clockwise fashion are: main starting menu (A), RNA secondary structure cluster trees (B), dot-matrix histogram (C), Stem Trace control window (D), a Stem Trace with multiple solution space bins (E), secondary structure drawing with samples of the system annotation capabilities (F), and a drawing of a very complex structure of 9229 nt long HIV-1 LAI (G).

the existence of competing populations. In the analysis of results we look for large secondary peaks indicating strong structural alternatives. We also check if for a given population the dominant results correspond to the best fitness (lowest free energy) peaks, and, if not, we see it as an indication that higher population runs are necessary to look for the best fitness structures.

(6) Max., min. and avg. files—These files contain data on the minimum, maximum and average energies for each generation of each run.

(7) Motif and stemtable—These files can be generated after initialization and contain all the stems that are used by the genetic algorithm's operators (the alphabet). The motif table specifically contains those stems that make up additional constructs that are considered to be motifs. This currently includes coaxially stacked stems. These tables may be modified and preloaded into MPGAfold for experimental purposes bypassing the initialization procedures that originally generated these files.

(8) Trace—A file containing for each generation the number of occurrences of a pre-specified stem or group of stems in the population is produced. This file contains information that is similar in concept to the visual trace map mentioned above, and it is useful for determining the occurrence rates of different stems of interest.

2.5. Visual data mining with StructureLab

StructureLab is an extensive RNA/DNA structure analysis workbench, which controls activation of various algorithms running on many different computer platforms (SGIs, SUNs, HP Alphas, and PCs under Linux). The implementation aspects of the system were previously described in [40]. It was also mentioned briefly in publications when it was applied to solving specific problems, such as described in [1,43,44]. It functions in a client/server mode where, in some cases a tight coupling exists between the running remote program and the workbench, while in other cases a remote program may be run as a batch job while users can utilize other tools as they are waiting for the batch jobs to complete. The system is being constantly modified to the current research needs and has been significantly enhanced. Some of the StructureLab's graphical analysis tools are illustrated in Fig. 4. Most of the items depicted in it will be described in more detail in the following sections. Other capabilities include sequence operations, switching between different free energy tables [20,52–55] and interactively calculating free energies of secondary structures or structural motifs within them, interfaces to folding programs, linear motif matching, structure alignment, secondary structure drawing and manipulation, and generation of preliminary three-dimensional coordinates for a given secondary structure. The system can

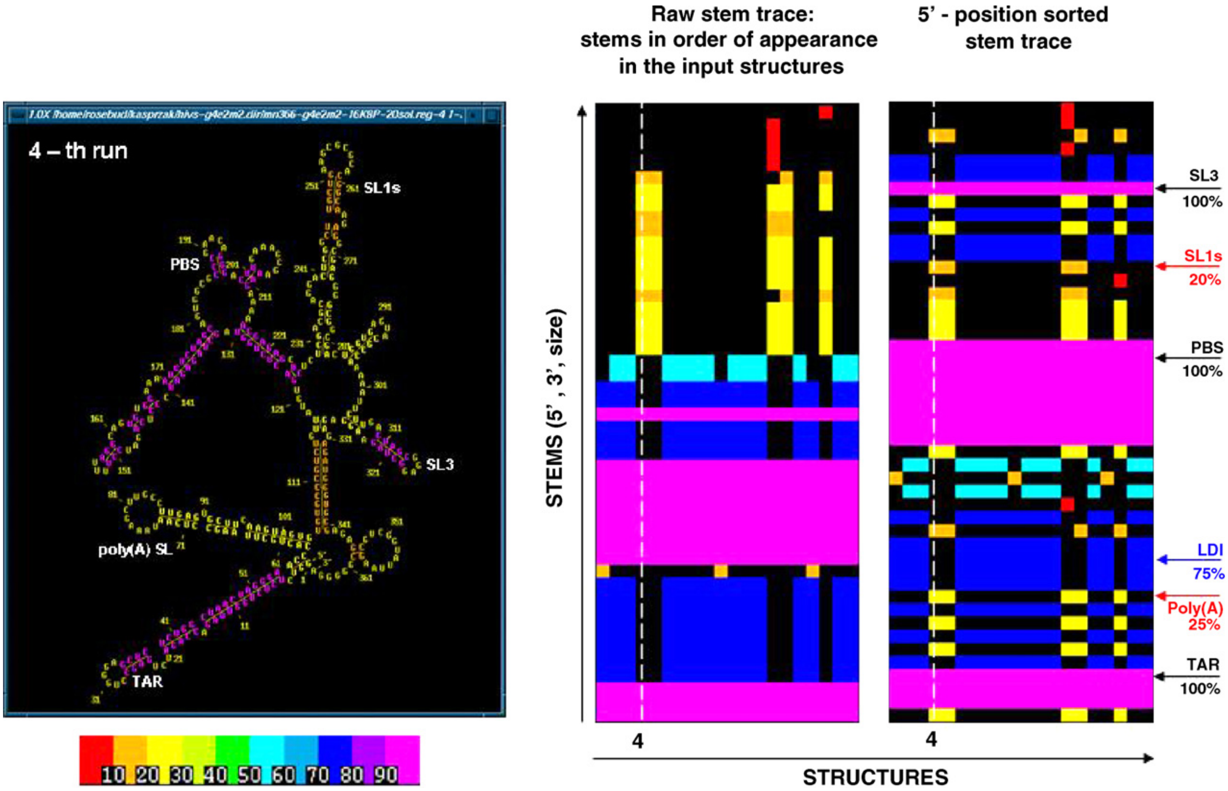


Fig. 5. Example of Stem Trace plots and a corresponding 2D structure related to the HIV-1 5′ UTR structure example discussed in the text. The two right-hand panels represent the same Stem Trace; the left one is unsorted (order of appearance of stems in the input structures) and the right one is 5′-position sorted. Structure 4 of the BMH (branched) topology type, indicated by the vertical dashed lines over the Stem Trace plots, is shown in the secondary structure drawing. Occurrence frequency of stems is color-coded (red means low occurrence and purple is high occurrence). The depicted data corresponds to final results of 20 MPGAfold runs in HIV-1 MN in its 5′ 366 nt domain, and it includes the low frequency (25%) BMH (branched) structures and the high frequency (75%) LDI (linear) conformers. Key stems are indicated for the 5′ sorted Stem Trace. Stems annotated in red belong to the BMH structures. Those annotated in blue are from the LDI conformations, and the black annotations indicate stems common to both conformation types.

retrieve sequences from a database, generate random mutations within specified boundaries, or splice them, and translate nucleic acid sequences to amino acid sequences. The integrated nature of many of the tools in this system allows the user to label a secondary structure drawing with the stems which are predicted to be stacked by the energy calculator [20]. Free energies of secondary structures can also be plotted. Selected sequences can be passed to remote programs (MPGAfold or Mfold) for folding (see Section 2.6 for more on StructureLab-MPGAfold interface). The predicted secondary structures can be drawn and interactively manipulated in a variety of ways. The system also has an interface to the RNA_2D3D package, [H. Martinez, J. Maizel Jr., B.A. Shapiro, manuscript in preparation], which has already proven to be a valuable tool in 3D structural studies and is briefly described in [56]. When invoked from StructureLab it is used in its "background" (non-interactive) mode to calculate preliminary 3D coordinates, with possible refinements via the AMBER [57] or Tinker [http://dasher.wustl.edu/tinker/] packages. The generated 3D coordinates file (PDB format) is automatically passed to a visualization program such as RasMol (by default) [58,59]. The remainder of this paper will focus on how StructureLab tools can be applied to the analysis of the large amounts of data that are produced by the massively parallel genetic algorithm for RNA structure prediction. We will emphasize the integration of many of the new features of StructureLab to determine the structures associated with the example folding of HIV-1 sequences.

2.5.1. Stem Trace

Stem Trace is a multifaceted visualization interface that is part of StructureLab. Originally described in [41], it has been significantly enhanced since then. It is defined as a two-dimensional graph. Each position along the y-axis represents a unique stem, defined by a triplet (5′ position, 3′ position, number of base pairs), from a secondary structure. The set of points intersected by a vertical line at position x in the graph represents all the stems that determine a secondary structure. Thus each position along the x-axis represents a structure, which can correspond to a generation from an MPGAfold run or a suboptimal structure produced by a DPA-based program. The default, raw ordering allocates consecutive y-axis positions to stems based on their "order of first appearance" in the input files. The 5′-sorted presentation has the advantage of depicting close to each other in the plot the stems which may be also topologically close in a secondary structure. Fig. 5 compares the above two orders side by side for the same MPGAfold data (HIV-1 MN) and shows a sample BMH (branched) type secondary structure drawing corresponding to the fourth structure plotted. Note that the majority of the solutions in this plot are of the LDI (linear) conformation type. Other available orderings sort the plotted stems by their 3′-order, length of stems, distance between the 5′ and 3′ ends, or their free energy. The "order of first appearance" Stem Traces are shown in Fig. 6 for an individual run of the genetic algorithm, from the first to the last generation, and in Fig. 7A for the final results of multiple population runs. This trace

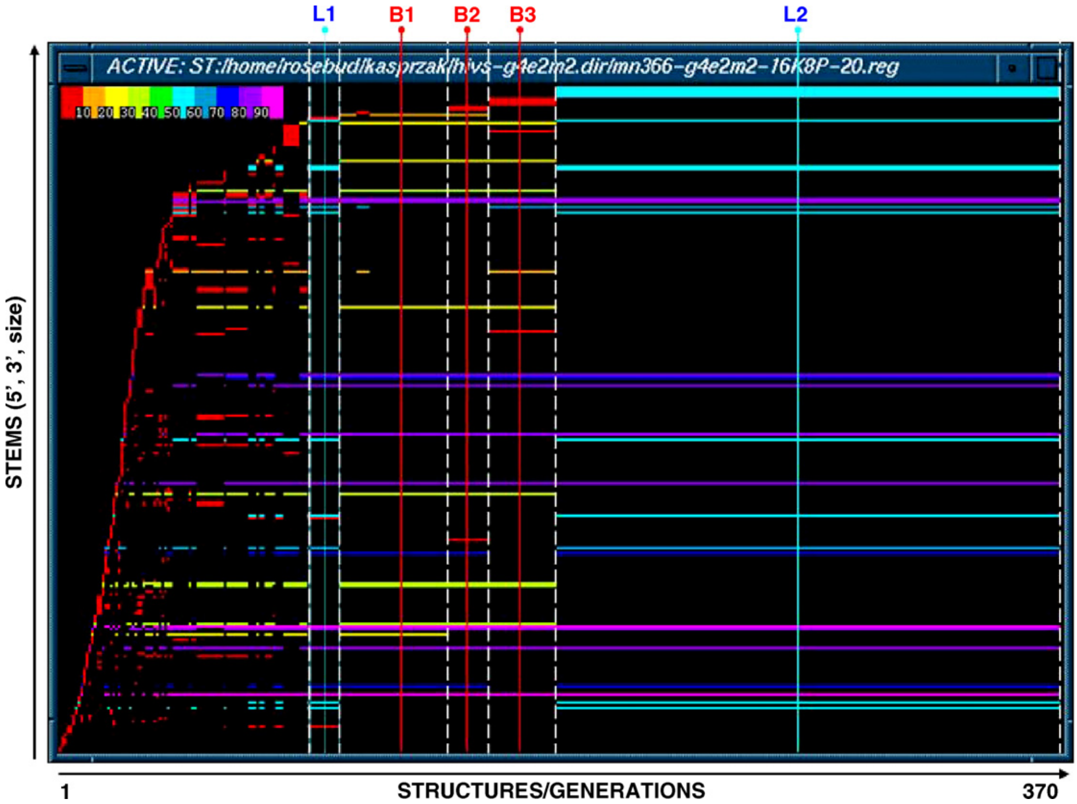
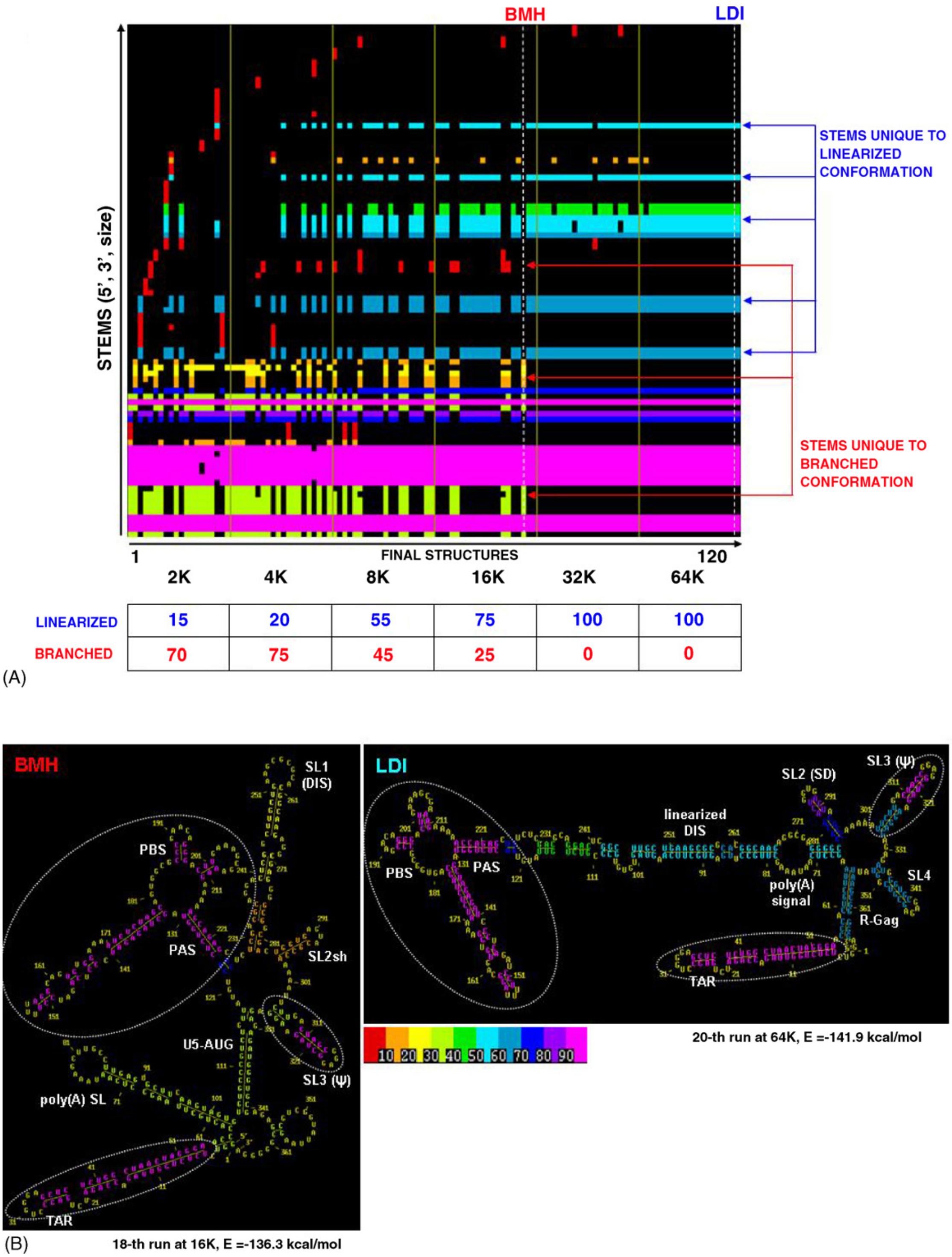


Fig. 6. A sample population consensus (histogram peak) Stem Trace depicting the maturation of the HIV-1 MN structure in one 16 K population MPGAfold run, 370 generations long. The order of stems along the y-axis reflects their order of appearance in the consecutive structures depicted along the x-axis. Labels, L for LDI (linear) and B for BMH (branched) conformations, indicate maturing structures of the two dominant HIV-1 5′ non-coding region topologies (see Fig. 7B) with variations. Stem frequency is color-coded, ranging from red (low) to purple (high).

ordering is particularly good in depicting structural motifs that enter a structure at the same time, which is clearly visible in the folding pathway captured in the Stem Trace shown in Fig. 6, where the BMH (marked B) and LDI (marked L) conformers introduce blocks of new stems to the plot.

In all cases stems are color-coded based upon frequency of occurrence, red being low frequency (up to 10%) and purple being high frequency (greater than 90%). Sets of structures can be placed in bins delineated by light vertical lines, as shown in Fig. 7A, showing either population variation results



	2K	4K	8K	16K	32K	64K
LINEARIZED	15	20	55	75	100	100
BRANCHED	70	75	45	25	0	0

Fig. 7. Relationship between a Stem Trace plot and sample secondary structures. (A) Example of a Stem Trace from multi-population MPGAfold runs. Each block represents final results of 20 runs for the control region of HIV-1 MN (366 nt long). The populations range from 2 to 64 K. Structural transitions are observable with increasing population sizes. Percentages of BMH (branched) and LDI (linear) structures in each population's solution spaces (bins) are listed in the table below the plot. (B) The dominant BMH and LDI structures drawn from the Stem Trace plot shown in (A) Structural motifs enclosed in dotted line ovals represent the ''scaffold'' motifs that do not undergo any structural changes.

(each bin contains results for different MPGAfold population runs) or multiple sequence family results (each bin contains the results from a given sequence). In the case of multiple solution space plots the stems can be color-coded based on the individual bin frequencies (i.e. separately for each solution bin) or total frequencies calculated for all solution bins. Free energy of stems can also be used as a color-coding criterion.

Stem Trace permits the depiction of many types of results generated by MPGAfold runs for both sequential and full length folds. Several of its capabilities, with illustrations depicting full length fold results, are enumerated below:

(1) Depiction of the maturation process of structures from an individual run of the genetic algorithm, from the first to the last generation—Raw output from an MPGAfold run can show the maturation of a structure over several generations. This is illustrated in Fig. 6. The lower generation numbers show few points (stems) while higher generations, usually contain more points indicating more mature and complex structures. Since in the genetic algorithm multiple motifs can enter a maturing structure simultaneously, several subdomains of the given input sequence can also fold at the same time, thus simulating multiple folding nucleation points. Stems that act as nucleation points will usually appear early in the formation of a motif and persist. This process can be especially evident in Stem Trace plots of sequential folding simulations.

(2) Depiction of the intermediate structures (consensus or best) from all runs of MPGAfold for a given sequence—This permits the visualization of the occurrence of common stem formation across multiple runs.

(3) Depiction of the final structures generated by MPGAfold—These structures correspond to representatives of the majority of the population of structures or the best fit final structures in the population (Figs. 5 and 7A).

(4) Depiction of the final structures generated by the genetic algorithm from several different sequences that comprise a family—A multiple sequence alignment is required in this case to present the stems of the structures in their proper positions [41]. This idea is illustrated in Fig. 8 for a trace plot of single stranded regions, derived from a Stem Trace of strains HIV-1 LAI and MN (see the next section for more details).

(5) Depiction of population variation runs—This permits the portrayal, for example, of populations ranging from 2 to 64 K, etc., all in one presentation. This allows structural comparisons for detecting metastable states that may transition to more stable states in higher population runs. Shown in Fig. 7A is a combined presentation of 2 K through 64 K runs for the control region of HIV-1 MN. Illustrated here is the transition from the BMH (branched) conformation, at lower populations, to the LDI (linear) conformation at higher populations [4,42–48]. However, certain stems are maintained in both conformations (dark blue and purple). This type of representation is very useful for the detection of metastable states as in the case presented in this example. Fig. 7B shows the two dominant conformations (BMH and LDI) with the regions in dotted-line ovals indicating constant motifs.

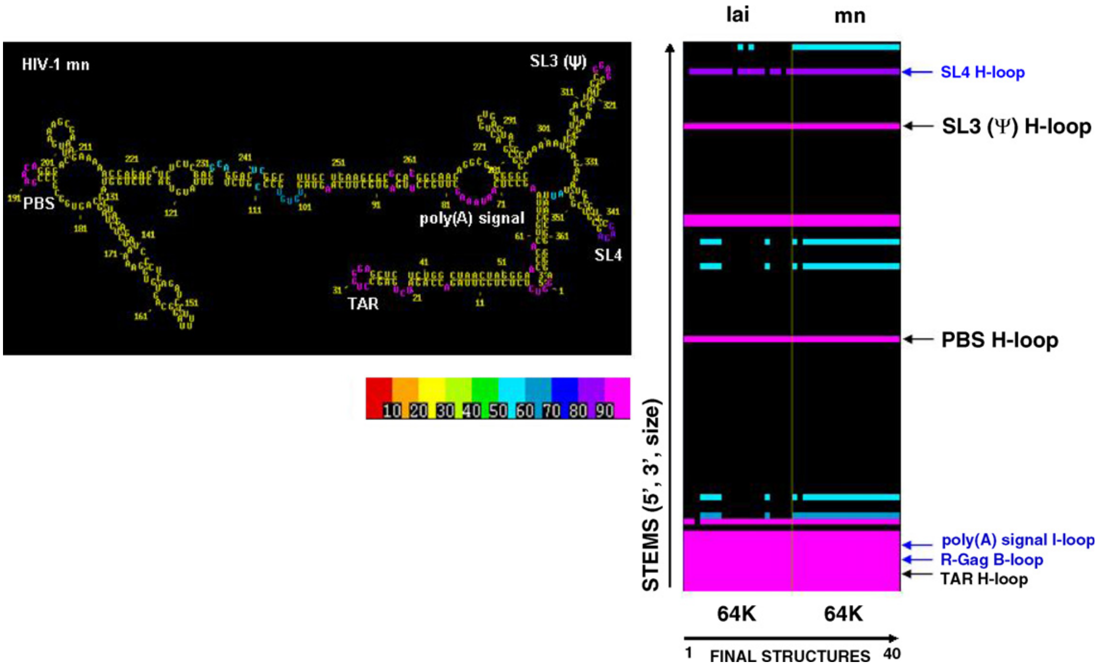
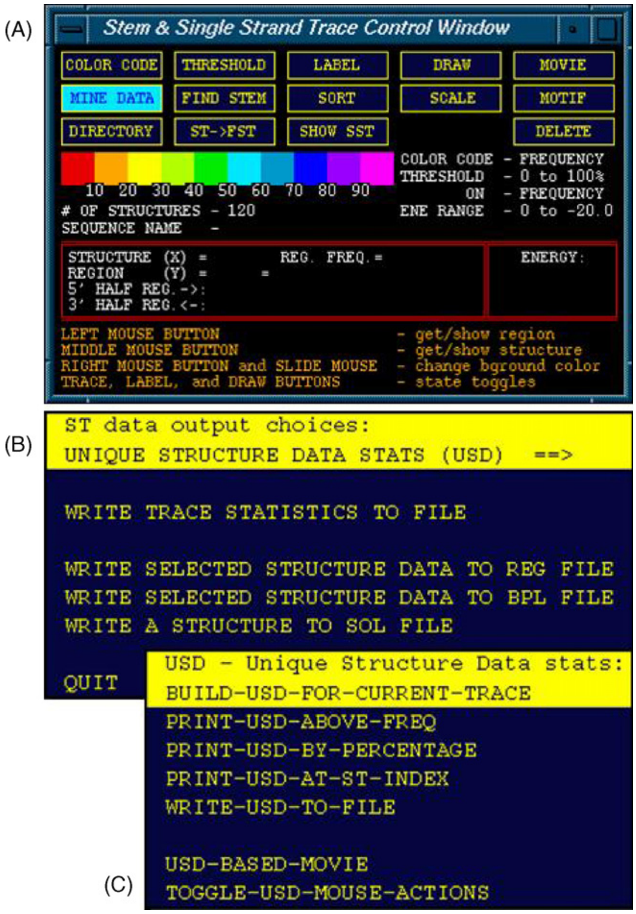


Fig. 8. Illustration of Single Strand Trace for the LDI (linear) HIV-1 MN and HIV-1 LAI conformers. For each strain, the final results of 20 MPGAfold runs with 64 K populations are shown. The results have been filtered for those single-stranded regions predicted in more than 50% of runs. Merging the Stem Traces for both sequences permits a structural comparison for the conservation, in this case, of single-stranded regions. The secondary structure shown (LDI) has the single strands corresponding to those displayed in the plot color-coded based on their frequency of occurrence in both HIV-1 strains.

2.5.2. Stem Trace control window

The Stem Trace control window (Figs. 4D and 9A) provides access to controls and analysis function buttons, as well as the display of information related to a Stem Trace plot. A mouse pointer may be moved over the Stem Trace plot causing specific information associated with individual stems to be displayed in the window. In general, left mouse button clicks extract information for one stem (at position x, y) and apply the selected function to it, while middle button clicks apply selected functions to the entire set of stems (one secondary structure) at position x. The most important functions are described below:

(1) *Mine Data*—Two levels of sub-menus present data capture and output options (Fig. 9). Statistics derived from the data depicted in a Stem Trace plot, cumulative and/or divided into individual solution space bins, in the case of multiple population or multiple sequence data sets, can be stored in a file. The new unique structure data (USD) sub-menu facilitates identification and examination of occurrences of all unique structures within in a given Stem Trace plot. Thus, for example, (Fig. 10) shows how in a plot of a single MPGAfold run, one can quickly determine and visualize the number of the most frequent unique structures and their occurrence rates. This type of data can be output as a statistics listing, as shown in Fig. 10 for the two dominant clusters of the HIV-1 MN conformers, the LDI (linear) and the BMH (branched) structures. The unique structure data can also be viewed in the form of a movie sequence of the representative secondary structures (see 6 below), and it is useful for determining the existence of folding pathway states within a run.



COLOR CODE - FREQUENCY
THRESHOLD - 0 to 100%
ON - FREQUENCY
ENE RANGE - 0 to -20.0
OF STRUCTURES - 120
SEQUENCE NAME -

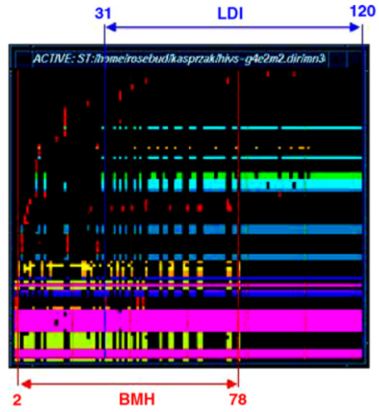
LEFT MOUSE BUTTON - get/show region
MIDDLE MOUSE BUTTON - get/show structure
RIGHT MOUSE BUTTON and SLIDE MOUSE - change bground color
TRACE, LABEL, and DRAW BUTTONS - state toggles

Fig. 9. Stem Trace Control Window (A) with two levels of submenus listing available data mining functions (B) and specific unique structure data (USD) creation, retrieval and interaction functions (C).

(2) *Single Strand Trace* (*SST*)—A plot complementary to the Stem Trace may be generated which shows the occurrence rate of the single stranded regions (a regular Stem Trace depicts the occurrence rate of helical base paired regions). This new feature could be used to compare results of the secondary structure predictions with available experimental structure probing data such as, for example, the novel probing technique called SHAPE, which appears to be particularly accurate in determining unpaired nucleotides [10,60]. In terms of the solution space analyses the SST is especially useful for depiction of stable loops associated with stems of variable length in folding solutions, such as the constant SL1 hairpin loop, which was associated in the predicted BMH structures with two variants of the stem: SL1 (Fig. 7B) and SL1s (Fig. 5). Fig. 8 illustrates the single strand trace plot being used for a comparison between two strains of HIV-1, MN and LAI, for the LDI (linear) conformation. A threshold of 50% is used to eliminate some noisy loop predictions. Sequence alignment procedures are used whenever more than one sequence is being compared to adjust for base insertions and deletions, and the corresponding trace plots are adjusted accordingly.

(3) *Motif*—Motif descriptors that define secondary structure topologies can be constructed and used to search for sequences which can accommodate them. This is similar to the RNAMOT program [61] except that the motif descriptors can be generated interactively and dynamically. The topology of the structure is built by selecting individual stems from the Stem Trace plot with a mouse. This topology can also be generated with a fuzzy characteristic allowing stems or single stranded regions to be of varying sizes. The descriptors can be written out into RNAMOT formatted files or be utilized in an internal search procedure, similar to RNAMOT, looking for sequences that can be constraint-folded.

(4) *Fuzzy Stem Trace*—When depicting the results from multiple runs of the genetic algorithm, whether it is with multiple strains from a family, or multiple results from one sequence, the Stem Trace plots may have to be adjusted to accommodate for slight differences in stem positions. The Fuzzy Stem Trace generator allows the user to interactively pick a set of reference stems from the current plot, similar to the way structural motifs are built (see 3 above). Next, it searches the space of solutions for stems that differ by 1 in their 5′ start or 3′ stop positions and, in addition, differ in size from the selected stems. Finally, a new plot is generated, in which all the stems matching a reference stem from the set of references are collapsed into one representative y-axis entry.



Dominant Linearized Structure (LDI)

```
GENERATION/STRUCTURE(X) = 111
REGIONS (UNADJUSTED POS.): ((5 54 11 -17.7) (17 43 5 -9.9) (25 38 4 -7.9)
                            (56 365 3 -6.6) (60 362 7 -11.2) (68 278 5 -10.3)
                            (80 267 6 -11.8) (87 260 2 -2.1) (90 257 8 -14.8)
                            (99 249 4 -4.1) (108 245 3 -6.7) (112 240 4 -6.7)
                            (116 233 3 -3.5) (123 226 2 -2.4) (125 223 7 -14.2)
                            (133 179 9 -16.9) (143 167 2 -3.3) (146 164 2 -2.1)
                            (151 159 3 -5.4) (187 197 3 -6.6) (199 212 3 -3.0)
                            (279 297 4 -9.1) (283 292 3 -4.3) (303 329 5 -5.7)
                            (310 323 5 -8.9) (334 353 2 -1.4) (337 350 5 -5.0))
ENERGY = -113.8 kcal
FIRST INDEX: 31
LAST INDEX: 120
FREQUENCY: 53/120 = 0.442
```

Dominant Branched Structure (BMH)

```
GENERATION/STRUCTURE(X) = 78
REGIONS (UNADJUSTED POS.): ((1 57 3 -5.5) (5 54 11 -17.7) (17 43 5 -9.9)
                            (25 38 4 -7.9) (58 104 8 -12.9) (66 95 4 -6.4)
                            (70 90 5 -7.5) (105 342 11 -20.0) (117 330 2 -1.3)
                            (123 226 2 -2.4) (125 223 7 -14.2) (133 179 9 -16.9)
                            (143 167 2 -3.3) (146 164 2 -2.1) (151 159 3 -5.4)
                            (187 197 3 -6.6) (199 212 3 -3.0) (229 283 3 -4.8)
                            (233 279 3 -5.8) (240 274 4 -6.9) (245 267 7 -12.0)
                            (285 298 5 -9.3) (305 327 3 -2.7) (310 323 5 -8.9)
                            (346 360 2 -3.4))
ENERGY = -109.4 kcal
FIRST INDEX: 2
LAST INDEX: 78
FREQUENCY: 10/120 = 0.083
```

Fig. 10. An example of data produced by the unique structure data mining function. A small copy of the Stem Trace previously shown in Fig. 7A is annotated to show the correspondences between the output listings and the plot. The output includes region lists, ranges and frequencies for two groups of unique secondary structures. They were selected to show the dominant conformation types corresponding to the BMH (branched) and LDI (linear) structures illustrated in Fig. 7B. The BMH structures identical to that selected at position 78 occur between positions 2 (MPGAfold 2 K population run) and 78 (MPGAfold 16 K population run) with a cumulative frequency (for all solution bins) of 8.3%. The LDI structures identical to that selected at position 111 occur between positions 31 (MPGAfold 4 K population run) and 120 (MPGAfold 64 K population run) with a cumulative frequency of 44.2%. Small structural differences in the BMH conformers account for the relatively low frequencies of the individual unique BMH structures. As a cluster of conformations, all the similar BMH structures are better captured in the topology taxonomy tree shown in Fig. 13B.

(5) *Draw/Label*—If the draw state button is toggled (Fig. 9A) in the control window, a mouse click over the Stem Trace plot invokes a drawing of a secondary structure containing the stems at position *x*. It is also possible to "point and click" at individual stems or structures (depending on the mouse button used) in the plot to have the corresponding regions in the structure drawing labeled in the color denoting the frequency of occurrence or the free energy of the selected stem. The color-coding of the structure drawing therefore allows one to associate the frequency of occurrence of the stems in a structure with the folding results. Figs. 5, 7A and B show typical Stem Trace plots together with color-coded secondary structures corresponding to the indicated *x*-positions in the plots.

(6) *Movie*—A movie consisting of the structures plotted in a given Stem Trace can be produced. Each structure is shown in sequence, paced either automatically or interactively by the user. A fixed stride can also be defined. The structures can be drawn in a standard form or as circle diagrams. If the Stem Trace plot represents a single run from the genetic algorithm, the movie will portray the developing structure from its immature state to its final state. On the other hand if

the Stem Trace depicts the final results of several runs or the final results of a population variation run, the variability of the final results will be portrayed.

(7) *Threshold*—The Stem Traces can be thresholded, reducing the data clutter by indicating only those stems that have a frequency of occurrence or free energies within a threshold range. When the draw option is selected with the thresholded Stem Trace, only those stems that survive the thresholding will be shown in the drawn secondary structure. A similar functionality is available for the Stem Histogram operator and will be illustrated below.

2.5.3. Stem histogram

There are times when one would like to view a database of secondary structures in a manner that is essentially orthogonal to that which is produced by Stem Trace. A Stem Histogram is a two-dimensional plot of possible base pair interactions from such a database, a concept also known from its other implementations as the dot matrix [62] or dot plot [19]. In this plot a given sequence is represented along the x-axis. The reverse complement of the sequence is represented along the y-axis and every potential base pair can be represented by a dot at any position in the matrix where there is a matching base. A database of possible structures can be loaded into a Stem Histogram where, if several structures have the same base pairs, they are color-coded indicating the frequency of occurrence of such pairs (based on the same color scale used elsewhere in StructureLab). Because equivalent stems from multiple input structures may differ in their 5′ start position, 3′ stop position or may not be of the same length, the corresponding diagonal runs of base pairs (stems) may be somewhat offset from each other, showing multiple colors (frequencies of occurrence) within what appears to be one diagonal (Fig. 11). The Stem Histogram is an interactive plot connected to the secondary structure drawing tools within StructureLab. The user can display information about base pairs by pointing and clicking at selected diagonals, as well as label secondary structures with nucleotides from a contiguous diagonal.

A threshold may be applied to the Stem Histogram, so that only the stems with a frequency of occurrence that surpasses the specified level are displayed. If the threshold is chosen above 50%, all the stems are compatible (non-conflicting), and a composite (consensus) structure may be drawn from all of them. This can be quite useful for depicting structural elements derived from genetic algorithm runs where fluctuations may occur in some portions of a conformation while other portions remain constant. Fig. 12 depicts the two conformational states of the HIV-1 MN 5′ non-coding region. The BMH (branched)

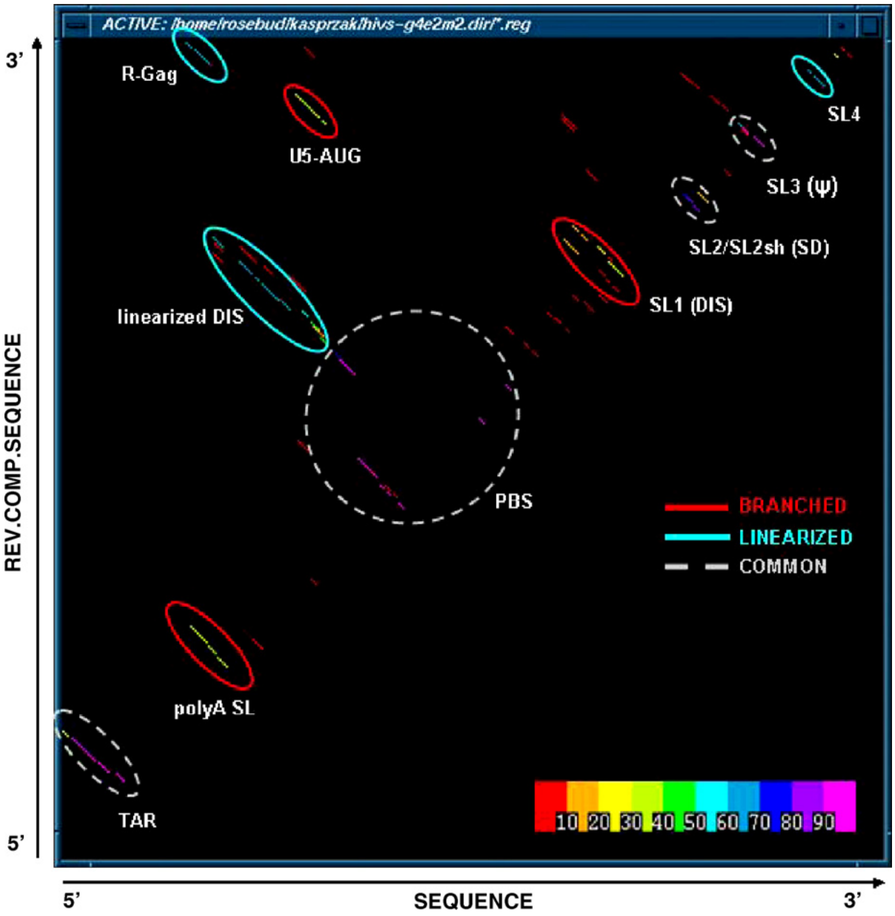


Fig. 11. Stem Histogram showing cumulative data for all base pairs from all MPGAfold populations runs of HIV-1 MN 366 nt domain. Stems that are constant in all the population runs are circled in light gray, those which are characteristic of the BMH (branched) conformations are circled in red, and the LDI (linear) conformation motifs are marked in blue. Note that the BMH and the LDI motifs are mutually exclusive.

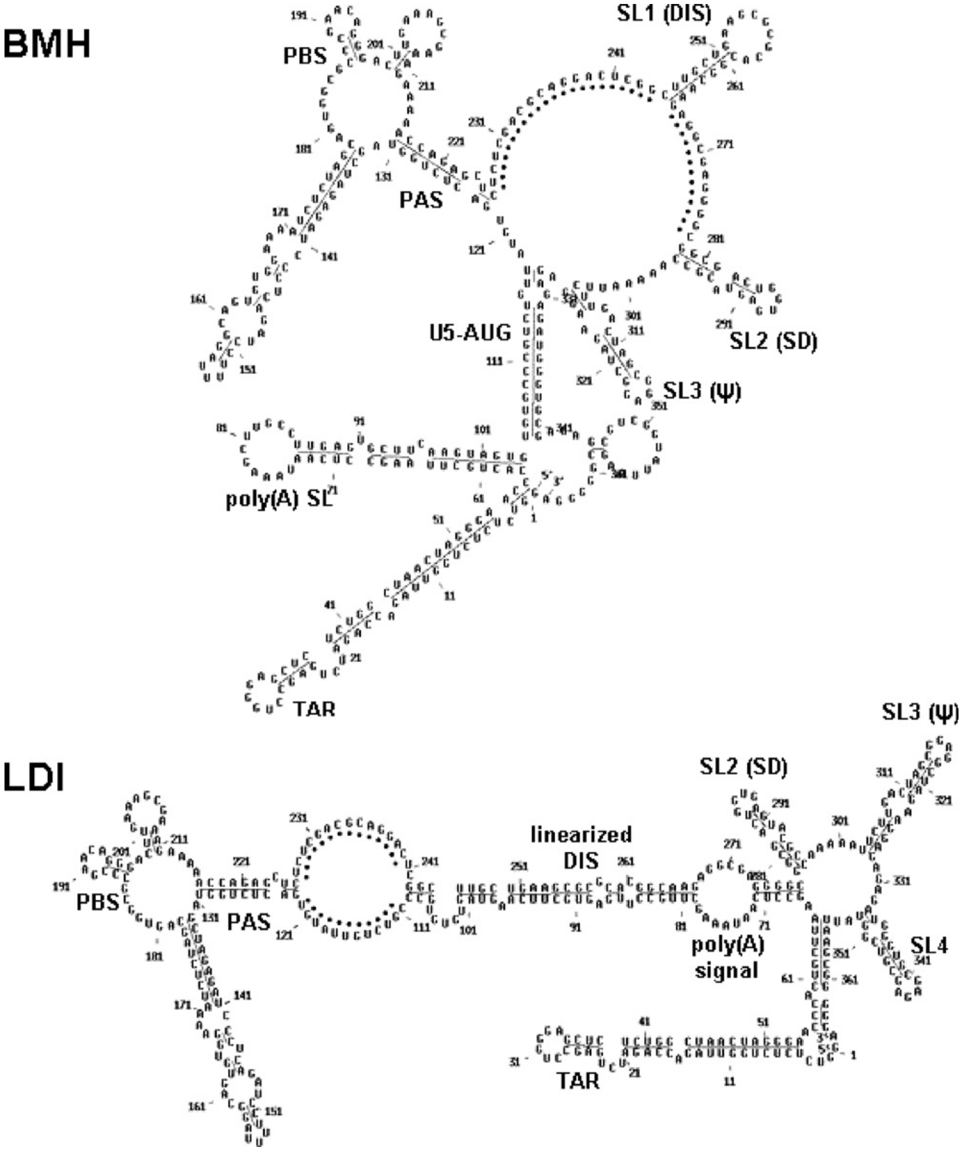


Fig. 12. Illustration of the composite structure capability of StructureLab. The two dominant structures containing stems that occur in more than 50% of final structures of multiple folding runs for HIV-1 MN (366 nt) are shown. BMH denotes the composite branched conformation based on 4 K population runs. LDI indicates the composite linear structure for the 16 K population runs. These two drawings capture the essence of the two conformations. Areas of variability that contain stems with less than a 50% occurrence rate are left open and are indicated with large dots.

composite structure was drawn from a Stem Histogram of multiple low-population MPGAfold final results, while the LDI conformer is a composite derived from a Stem Histogram of high population results. Single strand segments indicated with dotted lines correspond to regions of variability in the folding results.

The Stem Histogram data presentation is less sensitive to slight positional differences in stems when compared with Stem Trace, and it may indicate the magnitude of the differences more clearly since its positions are based on the absolute scale of sequence length. On the other hand, Stem Trace plots indicate clearly which stems are associated with which full structures in the examined database of predicted conformations. Compared with the Stem Histogram, the Stem Trace allows one to differentiate and examine the stems that appear in the different structures. Note that in our HIV-1 study example the strongly conserved ‘‘scaffold’’ motifs captured by the composite structures are also maintained in the final BMH and LDI structures representing MPGAfold's dominant low and high population conformations selected from a multi-population Stem Trace (Fig. 7B). In this respect the two plots allow the user to draw the same conclusions. However, if we were attempting to derive one composite structure from all the population runs data, the higher cumulative frequency of the LDI motifs, visible in the Fig. 11 Stem Histogram plot, would out-compete the BMH motifs. Careful examination of this histogram would reveal two mutually exclusive sets of stems, but this data is more striking in the Stem Trace presentation.

2.5.4. Taxonomy cluster trees

RNA secondary structures can be represented by a tree structure (Fig. 13A) and such a representation used in alignments of multiple conformations [63]. Using the concept

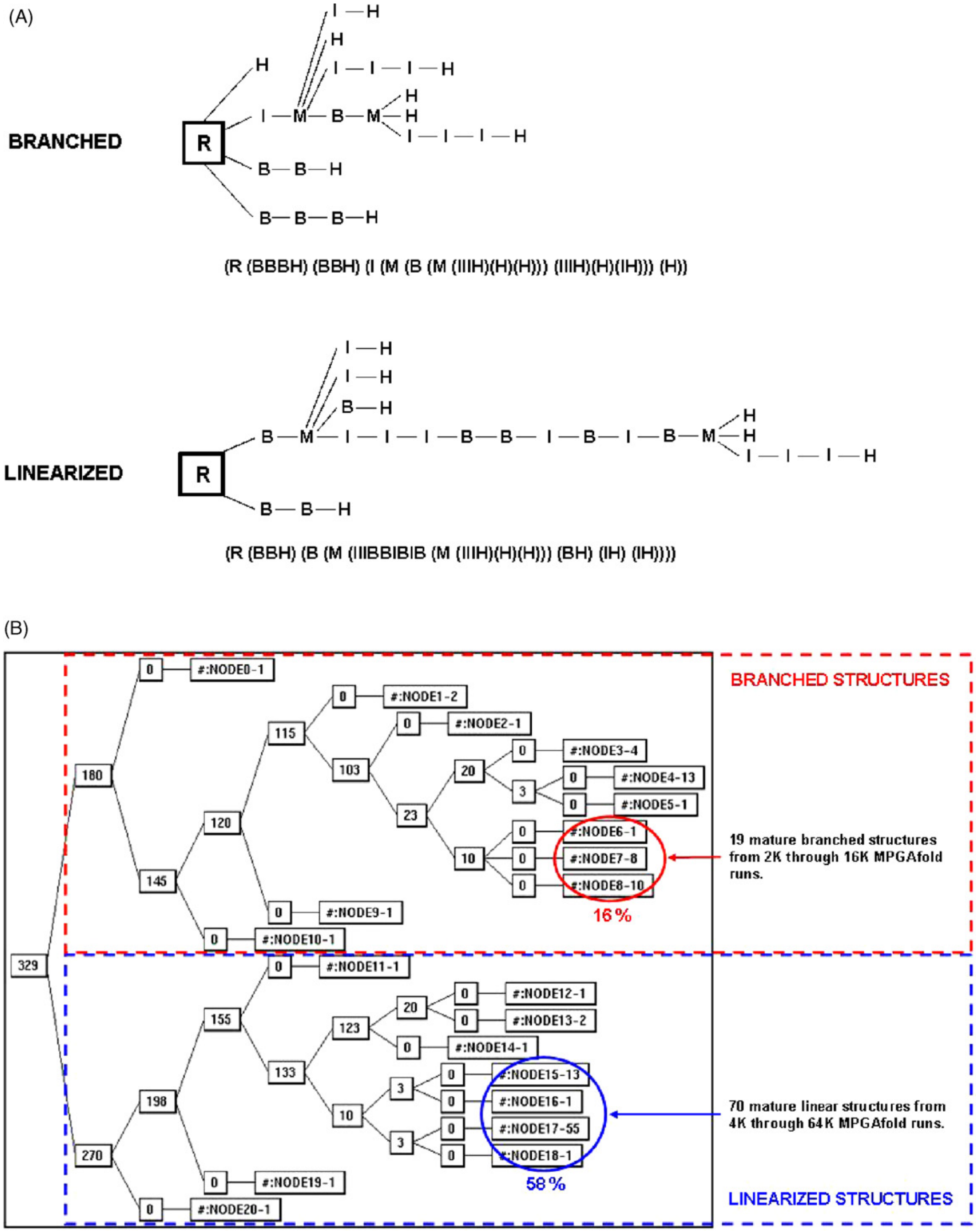


Fig. 13. (A) Tree representations of the topologies of the BMH (branched) and LDI (linear) structures of HIV-1 MN. The tree structures can also be represented by the parenthesis notation shown below them. R denotes the root node that ties together the entire tree. (B) Cluster tree derived from a tree-matching algorithm using topological trees similar to those shown in (A). The topological trees were generated from the results of all the MPGAfold population runs for HIV-1 MN. The terminal nodes, as indicated, are shown in a condensed mode depicting only the number of individual structures represented by that node rather than the names of each of the individual structures. This feature is used to conserve display space. It should be noted that two distinct clusters form, one representing the BMH (branched) conformation and the other representing the LDI (linear) conformation. Secondary structures illustrated in Fig. 7B are representative of conformations from NODE8-10 (BMH structure) and NODE17-55 (LDI structure), where 10 and 55 indicate the number of structures classified as identical (edit distance of 0) and represented by the respective nodes.

of a tree representation, a tree matching algorithm, which uses the tree edit distance can be applied to a database of structures generated by the genetic algorithm. Ultimately, a matrix of scores is produced which can be presented to a clustering algorithm to place together the similar secondary structures and to place apart dissimilar ones. The clustering, which was described in [64], has been implemented as a stand-alone program linked with StructureLab workbench. The same concept has been independently implemented in the Vienna package [65].

Three different levels of abstraction can be used to measure the degrees of similarity/dissimilarity. The first level simply measures the similarity based upon the topology of the structures. Thus, the edit distances are determined by the existence or non-existence of the topological features such as bulge loops, hairpin loops, internal loops and multibranch loops. The next level of abstraction considers the size of the loops and stems and adds the absolute size differences into the edit distance calculation. The third level takes into account the size differences inherent in the component parts of loops. Thus, the sum of the absolute differences of the component parts of the loop structures is factored into the cost. If the number of components differs, the best sum is chosen. Fig. 13B shows a taxonomy cluster tree drawn by StructureLab for the HIV-1 MN multiple-population data presented in Fig. 7A. Identical leaf nodes are collapsed into single representative nodes to compress the size of the cluster tree. The upper set of nodes is indicative of structures that are ‘‘branched’’ (BMH with variations) while the lower set of nodes represents those structures that are ‘‘linear’’ (LDI with variations).

2.6. Integration of MPGAfold and StructureLab

MPGAfold has been tied together with StructureLab by the use of a recently developed Java-based visualizer program (see Fig. 3), which allows the user to set MPGAfold parameters, invoke and communicate with this program, control the display of the population dynamics maps (described earlier) and also interact with StructureLab. It can also be used independently of StructureLab. Our analysis workbench can invoke the visualizer, which in turn activates and establishes a connection with MPGAfold. As MPGAfold is running, under the synchronized control of StructureLab, data is passed back to StructureLab for visualizing the current best fit structure, the consensus structure, or the structure from a user-specified population element in the evolving set of structures in the population. In addition, the genetic algorithm can be paused at any desired generation, and by pointing and clicking with a mouse on any pixel representing a structure in any of the three MPGAfold visualization maps, the structure present in that location can be displayed by StructureLab. Global drawing parameters set in StructureLab control the display of the structure as it is being drawn. For example, automatic untangling can be applied as well as labeling of stems which are being traced in the population. Such an interconnection makes it relatively easy to interactively explore evolving elements in a population.

3. Conclusion

The issue of RNA structure/function determination is a difficult and complex problem. The basic principle that is applied is that given an RNA sequence, the three-dimensional structure that is ultimately formed is completely determined by its sequence and its surrounding environment. Environmental factors such and solvent, ions and proteins are significant for structure/function determination. Thus, the ability to incorporate information about the external environment is important. The genetic algorithm allows some of this information to be used to guide the folding. For example, one can define a set of ‘‘sticky stems,’’ which act as hints during a run of MPGAfold, biasing the formation of specific structural elements. RNA secondary structures are important initial building blocks for the final three-dimensional structures. Also, it is important to remember that the final structure is not necessarily the only determinant of an RNA’s function. Functional intermediates or multiple stable states may occur that impart multiple functionality to these molecules. This is illustrated in the presentation for the case of the HIV-1’s untranslated region. The genetic algorithm (MPGAfold) is an example of a way to explore those states that may be important for RNA functionality. The examples of MPGAfold data visualizations presented in this paper indicate a key characteristic of the genetic algorithm; its ability to focus on significant conformations. This is strikingly revealed in the final population consensus results of multiple population runs, illustrated in the Stem Trace shown in Fig. 7A and in the single full run Stem Trace (Fig. 6). The MPGAfold’s population and trace maps (Fig. 3) clearly show two competing conformations which fill the whole solution space.

StructureLab used in conjunction with the genetic algorithm’s results has proven to be a very valuable tool for determining biologically functional states and illustrates the importance of interactive exploratory data analysis for the RNA folding problem. A vast amount of information is obtained from the genetic algorithm from both individual and multiple runs, including variable population runs. The various visualization tools that are part of StructureLab and MPGAfold permit the analysis of these results, as is illustrated by the examples of prediction of the two conformations of HIV-1 $5'$ non-coding region. The many perspectives provided by the tools discussed here are meant to complement each other, such as the Stem Trace and Stem Histogram compared and contrasted in the Stem Histogram section. In general, the Stem Trace representation of data makes it easier to identify clusters of full structures with a frequency of occurrence lower than 50%. This is particularly important when dealing with MPGAfold data, as it already focuses on significant conformations, which means that even lower frequency conformation clusters should not be ignored. Dealing with multiple conformational states requires multiple iterations in the analysis process. First, for example, we may want to identify topology-based clusters in the folding data by using the Taxonomy Tree tools. Next we may want to view all the high frequency unique structures produced by a folding algorithm via the data mining features of Stem Trace. Resolution of some noisy motif predictions can be obtained with the help of Stem Histogram. If the final results are not as striking as those presented in our figures, running the genetic algorithm interactively and tracing the dynamics of formation and rearrangements of the selected key motifs may be helpful in determining what other factors play a role in the folding process and if biasing MPGAfold runs with some structural hints suggested by experimental data, for example, could improve the clarity of the results.

Another illustration of graphical exploratory data analysis involves the combination of the movie display capability of Stem Trace with the sequential folding output of MPGAfold. This proved to be useful for interpreting the effects of co-transcriptional folding of an RNA virus. For example, viewing the results from several different individual sequential folding runs indicated two possible alternative folding pathways and indicated a critical point in the folding process, at which the two alternate pathways would diverge. One path showed the potential formation of a structure required for RNA replication, the other depicted the formation of a structure required for RNA editing. Both paths are required for the virus life-cycle.

In general, the order of analysis and the type of analysis one performs, once the folding data is generated, does not have to follow one rigid protocol. The above described examples illustrate some of the many possible approaches that can be applied. We also believe that they show the need for approaching the data from many perspectives, which requires flexible and interactive analysis tools.

It should also be noted that many of the visual data mining tools described in this paper can also be applied to the results of dynamic programming algorithms such as Mfold. For example, Stem Traces, Stem histograms (dot matrix), cluster trees, and the drawing facilities can all be employed with the output of these algorithms. Of course interpretation of the presented data has to be modified given the different paradigm under which they have been created.

Future research requires the integration of the visual data mining environment with sophisticated statistical tools to enable the extraction of subtle information such as correlated folding events and those specific structural elements that are important for folding pathways.

4. Availability

Programs mentioned here (MPGAfold, MPGAfold visualizer and StructureLab) and instructions for configuration on different computer architectures are available upon request. Please, contact Dr. Bruce Shapiro at bshapiro@ncifcrf.gov.

Acknowledgements

We wish to thank Dr. Hugo Martinez for providing his RNA_2D3D program and modifying it to our needs. We would like to acknowledge contribution of David Bengali to the initial work on MPGAfold's visualizer. This publication has been funded in whole or in part with Federal funds from the National Cancer Institute, National Institutes of Health, under Contract No. NO1-CO-12400.

This research was supported [in part] by the Intramural Research Program of the NIH, National Cancer Institute, Center for Cancer Research. The content of this publication does not necessarily reflect the views or policies of the Department of Health and Human Services, nor does mention of trade names, commercial products, or organizations imply endorsement by the U.S. Government.

References

[1] B.A. Shapiro, D. Bengali, W. Kasprzak, J.C. Wu, RNA folding pathway functional intermediates: their prediction and analysis, J. Mol. Biol. 312 (1) (2001) 27–44.

[2] K.S. Lavery, T.H. King, Antisense and RNAi: powerful tools in drug target discovery and validation, Curr. Opin. Drug Discov. Dev. 6 (4) (2003) 561–569.

[3] E. Nudler, A.S. Mironov, The riboswitch control of bacterial metabolism, Trends Biochem. Sci. 29 (1) (2004) 11–17.

[4] J.C. Paillart, M. Shehu-Xhilaga, R. Marquet, J. Mak, Dimerization of retroviral RNA genomes: an inseparable pair, Nat. Rev. Microbiol. 2 (6) (2004) 461–472.

[5] J.P. Staley, C. Guthrie, Mechanical devices of the spliceosome: motors, clocks, springs, and things, Cell 92 (3) (1998) 315–326.

[6] D.P. Giedroc, C.A. Theimer, P.L. Nixon, Structure, stability and function of RNA pseudoknots involved in stimulating ribosomal frameshifting, J. Mol. Biol. 298 (2) (2000) 167–185.

[7] E.A. Doherty, J.A. Doudna, Ribozyme structures and mechanisms, Annu. Rev. Biochem. 69 (2000) 597–615.

[8] D.H. Turner, N. Sugimoto, RNA structure prediction, Annu. Rev. Biophys. Biophys. Chem. 17 (1988) 167–192.

[9] H.F. Noller, Structure of ribosomal RNA, Annu. Rev. Biochem. 53 (1984) 119–162.

[10] E.J. Merino, K.A. Wilkinson, J.L. Coughlan, K.M. Weeks, RNA structure analysis at single nucleotide resolution by selective 2′-hydroxyl acylation and primer extension (SHAPE), J. Am. Chem. Soc. 127 (12) (2005) 4223–4231.

[11] M. Edelman, I.M. Verma, D. Saya, U.Z. Littauer, Optical absorbance properties of mitochondrial ribosomal RNA, Biochem. Biophys. Res. Commun. 42 (2) (1971) 208–213.

[12] D. Riesner, G. Steger, R. Zimmat, R.A. Owens, M. Wagenhofer, W. Hillen, S. Vollbach, K. Henco, Temperature-gradient gel electrophoresis of nucleic acids: analysis of conformational transitions, sequence variations, and protein–nucleic acid interactions, Electrophoresis 10 (5–6) (1989) 377–389.

[13] E. Bindewald, B.A. Shapiro, RNA secondary structure prediction from sequence alignments using a network of k-nearest neighbor classifiers, RNA 12 (3) (2006) 342–352.

[14] B. Knudsen, J. Hein, Pfold: RNA secondary structure prediction using stochastic context-free grammars, Nucl. Acids Res. 31 (13) (2003) 3423–3428.

[15] B. Knudsen, J. Hein, RNA secondary structure prediction using stochastic context-free grammars and evolutionary history, Bioinformatics 15 (6) (1999) 446–454.

[16] I.L. Hofacker, M. Fekete, P.F. Stadler, Secondary structure prediction for aligned RNA sequences, J. Mol. Biol. 319 (5) (2002) 1059–1066.

[17] J. Ruan, G.D. Stormo, W. Zhang, An iterated loop matching approach to the prediction of RNA secondary structures with pseudoknots, Bioinformatics 20 (1) (2004) 58–66.

[18] J. Ruan, G.D. Stormo, W. Zhang, ILM: a web server for predicting RNA secondary structures with pseudoknots, Nucl. Acids Res. 32 (Web Server issue) (2004) W146–W149.

[19] M. Zuker, On finding all suboptimal foldings of an RNA molecule, Science 244 (4900) (1989) 48–52.

[20] D.H. Mathews, J. Sabina, M. Zuker, D.H. Turner, Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure, J. Mol. Biol. 288 (5) (1999) 911–940.

[21] D.H. Mathews, M.D. Disney, J.L. Childs, S.J. Schroeder, M. Zuker, D.H. Turner, Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure, Proc. Natl. Acad. Sci. U.S.A. 101 (19) (2004) 7287–7292.

[22] I.L. Hofacker, W. Fontana, P.F. Stadler, S. Bonhoeffer, M. Tacker, P. Schuster, Fast folding and comparison of RNA secondary structures, Monatsh. Chem. 125 (1994) 167–188.

[23] E. Rivas, S.R. Eddy, A dynamic programming algorithm for RNA structure prediction including pseudoknots, J. Mol. Biol. 285 (5) (1999) 2053–2068.

[24] R.M. Dirks, N.A. Pierce, A partition function algorithm for nucleic acid secondary structure including pseudoknots, J. Comput. Chem. 24 (13) (2003) 1664–1677.

[25] J. Reeder, R. Giegerich, Design, implementation and evaluation of a practical pseudoknot folding algorithm based on thermodynamics, BMC Bioinformat. 5 (2004) 104.

[26] J. Ren, B. Rastegari, A. Condon, H.H. Hoos, HotKnots: heuristic prediction of RNA secondary structures including pseudoknots, RNA 11 (10) (2005) 1494–1504.

[27] R. Giegerich, B. Voss, M. Rehmsmeier, Abstract shapes of RNA, Nucl. Acids Res. 32 (16) (2004) 4843–4851.

[28] P. Steffen, B. Voss, M. Rehmsmeier, J. Reeder, R. Giegerich, RNAshapes: an integrated RNA analysis package based on abstract shapes, Bioinformatics 22 (4) (2006) 500–503.

[29] B. Voss, R. Giegerich, M. Rehmsmeier, Complete probabilistic analysis of RNA shapes, BMC Biol. 4 (1) (2006) 5.

[30] Y. Ding, C.Y. Chan, C.E. Lawrence, RNA secondary structure prediction by centroids in a Boltzmann weighted ensemble, RNA 11 (8) (2005) 1157–1166.

[31] C.Y. Chan, C.E. Lawrence, Y. Ding, Structure clustering features on the Sfold Web server, Bioinformatics 21 (20) (2005) 3926–3928.

[32] A. Xayaphoummine, T. Bucher, H. Isambert, Kinefold web server for RNA/DNA folding path and structure prediction including pseudoknots and knots, Nucl. Acids Res. 33 (Web Server issue) (2005) W605–W610.

[33] H. Isambert, E.D. Siggia, Modeling RNA folding paths with pseudoknots: application to hepatitis delta virus ribozyme, Proc. Natl. Acad. Sci. U.S.A. 97 (12) (2000) 6515–6520.

[34] B.A. Shapiro, J.C. Wu, An annealing mutation operator in the genetic algorithms for RNA folding, Comput. Appl. Biosci. 12 (3) (1996) 171–180.

[35] B.A. Shapiro, J. Navetta, A massively parallel genetic algorithm for RNA secondary structure prediction, J. Supercomput. 8 (1994) 195–207.

[36] B.A. Shapiro, J.C. Wu, Predicting RNA H-type pseudoknots with the massively parallel genetic algorithm, Comput. Appl. Biosci. 13 (4) (1997) 459–471.

[37] J.-C. Wu, B.A. Shapiro, A Boltzmann filter improves the prediction of RNA folding pathways in a massively parallel genetic algorithm, J. Biomol. Struct. Dyn. 17 (63) (1999) 581–595.

[38] B.A. Shapiro, J.C. Wu, D. Bengali, M.J. Potts, The massively parallel genetic algorithm for RNA folding: MIMD implementation and population variation, Bioinformatics 17 (2) (2001) 137–148.

[39] B. Shapiro, D. Bengali, W. Kasprzak, J.C. Wu, Computational insights into RNA folding pathways:Getting from here to there, in: The Proceedings of the Atlantic Symposium on Computational Biology, Genome Systems and Technology, 2001, pp. 10–13.

[40] B.A. Shapiro, W. Kasprzak, STRUCTURELAB: a heterogeneous bioinformatics system for RNA structure analysis, J. Mol. Graph. 14 (4) (1996), 194–205, 222–194.

[41] W. Kasprzak, B. Shapiro, Stem Trace: an interactive visual tool for comparative RNA structure analysis, Bioinformatics 15 (1) (1999) 16–31.

[42] W. Kasprzak, B.A. Shapiro, Structural dependencies of the HIV-1 dimer initiation site as determined by the massively parallel genetic algorithm, in: The Proceedings of the International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences, 2002, pp. 48–54.

[43] W. Kasprzak, E. Bindewald, B.A. Shapiro, Structural polymorphism of the HIV-1 leader region explored by computational methods, Nucl. Acids Res. 33 (22) (2005) 7151–7163.

[44] A.H. Gee, W. Kasprzak, B.A. Shapiro, Structural differentiation of the HIV-1 polyA signals, J. Biomol. Struct. Dyn. 23 (4) (2006) 417–428.

[45] B. Berkhout, M. Ooms, N. Beerens, H. Huthoff, E. Southern, K. Verhoef, In vitro evidence that the untranslated leader of the HIV-1 genome is an RNA checkpoint that regulates multiple functions through conformational changes, J. Biol. Chem. 277 (22) (2002) 19967–19975.

[46] H. Huthoff, B. Berkhout, Multiple secondary structure rearrangements during HIV-1 RNA dimerization, Biochemistry 41 (33) (2002) 10439–10445.

[47] H. Huthoff, B. Berkhout, Two alternating structures of the HIV-1 leader RNA, RNA 7 (1) (2001) 143–157.

[48] C.K. Damgaard, E.S. Andersen, B. Knudsen, J. Gorodkin, J. Kjems, RNA interactions in the 5′ region of the HIV-1 genome, J. Mol. Biol. 336 (2) (2004) 369–379.

[49] J.H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, MI, 1975.

[50] J.H. Holland, Adaptation in Natural and Artificial Systems: An Introductory Analysis with Aplications in Biology, Control, and Artificial Intelligenece, MIT Press, Cambridge, MA, 1992.

[51] G.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989.

[52] J.A. Jaeger, D.H. Turner, M. Zuker, Improved predictions of secondary structures for RNA, Proc. Natl. Acad. Sci. U.S.A. 86 (20) (1989) 7706–7710.

[53] J.A. Jaeger, D.H. Turner, M. Zuker, Predicting optimal and suboptimal secondary structure for RNA, Methods Enzymol. 183 (1990) 281–306.

[54] A.E. Walter, D.H. Turner, J. Kim, M.H. Lyttle, P. Muller, D.H. Mathews, M. Zuker, Coaxial stacking of helixes enhances binding of oligoribonucleotides and improves predictions of RNA folding, Proc. Natl. Acad. Sci. U.S.A. 91 (20) (1994) 9218–9222.

[55] S.M. Freier, R. Kierzek, J.A. Jaeger, N. Sugimoto, M.H. Caruthers, T. Neilson, D.H. Turner, Improved free-energy parameters for predictions of RNA duplex stability, Proc. Natl. Acad. Sci. U.S.A. 83 (24) (1986) 9373–9377.

[56] Y.G. Yingling, B.A. Shapiro, The prediction of the wild-type telomerase RNA pseudoknot structure and the pivotal role of the bulge in its formation, J. Mol. Graph. Model. 25 (2006) 261–274.

[57] D.A. Case, T.E. Cheatham 3rd, T. Darden, H. Gohlke, R. Luo, K.M. Merz Jr., A. Onufriev, C. Simmerling, B. Wang, R.J. Woods, The Amber biomolecular simulation programs, J. Comput. Chem. 26 (16) (2005) 1668–1688.

[58] R.A. Sayle, E.J. Milner-White, RASMOL: biomolecular graphics for all, Trends Biochem. Sci. 20 (9) (1995) 374.

[59] R.A. Sayle, RasMol v2.5. A Molecular Visualization Program. Middlesex, UK, 1994.

[60] K.A. Wilkinson, E.J. Merino, K.M. Weeks, RNA SHAPE chemistry reveals nonhierarchical interactions dominate equilibrium structural transitions in tRNA(Asp) transcripts, J. Am. Chem. Soc. 127 (13) (2005) 4659–4667.

[61] A. Laferriere, D. Gautheret, R. Cedergren, An RNA pattern matching program with enhanced performance and portability, Comput. Appl. Biosci. 10 (2) (1994) 211–212.

[62] J.V. Maizel Jr., R.P. Lenk, Enhanced graphic matrix analysis of nucleic acid and protein sequences, Proc. Natl. Acad. Sci. U.S.A. 78 (12) (1981) 7665–7669.

[63] B.A. Shapiro, An algorithm for comparing multiple RNA secondary structures, Comput. Appl. Biosci. 4 (3) (1988) 387–393.

[64] B.A. Shapiro, K.Z. Zhang, Comparing multiple RNA secondary structures using tree comparisons, Comput. Appl. Biosci. 6 (4) (1990) 309–318.

[65] I.L. Hofacker, Vienna RNA secondary structure server, Nucl. Acids Res. 31 (13) (2003) 3429–3431.