Modeling and Predicting RNA Three-dimensional Structures

Jérôme Waldispühl & Vladimir Reinharz

School of Computer Science, McGill University, Montreal, Canada.

Summary

Modeling the three-dimensional structure of RNAs is a milestone toward better understanding and prediction of nucleic acids molecular functions. Physics-based approaches and molecular dynamics simulations are not tractable on large molecules with all-atom models. To address this issue, coarse-grained models of RNA three-dimensional structures have been developed. In this chapter, we describe a graphical modeling based on the Leontis-Westhof extended base pair classification. This representation of RNA structures enables us to identify highly conserved structural motifs with complex nucleotide interactions in structure databases. Further, we show how to take advantage of this knowledge to quickly and simply predict three-dimensional structures of large RNA molecules.

Key words: tertiary structure, RNA motifs, extended secondary structure, base pair classification, modeling, prediction.

Introduction

RNAs perform a broad range of catalytic and regulatory functions in cells, which often use the folding properties of these molecules. RNA structures are typically described at two levels of organization. The secondary structure, which enumerates the Watson-Crick and Wobble base pairs that create the backbone of the molecular structure, and the tertiary structure, which indicates the positions of each atom in the molecule.

The study of secondary structures and base pairing properties can reveal fundamental insights into the functional mechanisms of RNAs such as frameshift elements (Bekaert, et al. 2003) and riboswitches (Vitreschak, et al. 2004). However, the information provided by this representation is sometimes not sufficient to describe the complexity of inter- and intra- molecular interactions that govern RNA functions. A seminal example is the sarcin-ricin factor-binding loop, which possesses a sophisticated tertiary structure (Szewczak, et al. 1993).

Classical approaches to predict and analyze three-dimensional molecular structures make use of molecular dynamic simulations (Šponer, et al. 2012). This methodology has the potential to perform calculations on all-atom models, but suffers from high computational complexity. Straightforward applications are thus limited to small molecules (approximately 50 nt.) on a short period of time with very small body motion. Coarse-grained modeling of RNA structures enables us to overcome some of these limitations (Ding, et al. 2008, Jonikas, et al. 2009, Bernauer, et al. 2011, Poursina, et al. 2011), but the practical impact and theoretical horizon of these technologies remain unchanged. In addition, molecular dynamic simulations often requires fine-tuning and can be challenging to run and interpret for non-experts.

Recently, several research groups developed alternate strategies to model and predict RNA three-dimensional structures. One of the most successful approach, applied in the MC-Fold|MC-Sym pipeline (Parisien and Major 2008), RNA2D3D (Martinez, Maizel and Shapiro 2008) or 3dRNA (Zhao, et al. 2012), consists in predicting a secondary structure first, and use it to build a three-dimensional model. Other programs have employed fragment-assembly (Das and Baker 2007, Das, Karanicolas and Baker 2010) or conditional random fields techniques (Wang and Xu 2011). It is worth noting that, as we saw in a previous chapter, comparative modeling techniques can also be applied to RNA molecules if one structural homolog has been already identified (Rother, et al. 2011).

In this chapter, we introduce a versatile model for RNA structures, which enables us to describe essential features and fine-grained details of RNA three-dimensional structures while preserving the complexity of the representation to a minimum. This methodological advance is key to large-scale analysis and to better understanding of critical features of RNA molecular structures. In turn, this knowledge enables us to define new computational techniques to quickly and easily predict three-dimensional structures of large RNA molecules.

First, we describe the base pair classification at the base of this model (Leontis and Westhof 2001), and present rnaview (Yang, et al. 2003) -- a program that annotates automatically RNA three-dimensional structures. Then, we introduce the concept of RNA structural motifs and present recent databases and online resources based on this definition (Djelloul and Denise 2008, Leontis and Zirbel 2012). Finally, show how to use this knowledge to predict RNA three-dimensional structures. The pipeline described in this chapter works in two steps. First, we predict secondary structures with RNAsubopt (Hofacker, et al. 1994) and expand the prediction to a full base-pairing interaction network with RNA-MoIP (Reinharz, Major and Waldispühl 2012). Next, we use this graphical representation to build the three-dimensional structure of the RNA with MC-Sym (Parisien and Major 2008).

Material

Sequence and structure data

The Nucleic Acids Database (Berman, et al. 1992) is a repository of experimentally determined three-dimensional RNA structures maintained at Rutgers University, which is available at http://ndbserver.rutgers.edu. This is a specialized version of the Protein Data Bank (Bernstein, et al. 1977), which is available at http://www.rcsb.org/pdb. The structures are typically stored in files with the extension ".pdb". In particular, the PDB files store the spatial coordinates of each atom in the molecule. Plain sequences can also be downloaded separately under the FASTA format.

In this chapter, we illustrate our methods on the sequence and experimentally determined tertiary structures of tRNA(Cys) of *Archaeoglobus fulgidus* (Fukunaga and Yokoyama 2007). The PDB ID of this molecule is 2DU3.

Automatic annotation of 3D structures

The rnaview software is used to identify all base-pairing interactions that represent in a RNA tertiary structure (Yang, et al. 2003). This program can be downloaded at: http://ndbserver.rutgers.edu/ndbmodule/services/download/rnaview.html and is currently available for Linux, UNIX, SUN, and MAC systems.

RNA secondary structure prediction tools

The prediction of RNA secondary structures is performed with the RNAfold program available in the Vienna RNA package (Hofacker, et al. 1994, Lorenz, et al. 2011). The software suite is available at http://www.tbi.univie.ac.at/RNA/ and can runs under LINUX, UNIX, MAC and WINDOWS systems. In this chapter, we used the version 2.1.2 of the Vienna RNA package. Web services are also available at http://rna.tbi.univie.ac.at/.

Insertion of RNA motifs in secondary structure

We perform the insertion of RNA motifs inside predicted RNA secondary structures with the RNA-MoIP software (Reinharz, Major and Waldispühl 2012) available at http://csb.cs.mcgill.ca/RNAMoIP/. Noticeably, this software requires installing the Gurobi solver (http://www.gurobi.com/), which is free for academic users.

Building RNA 3D structures from a RNA interaction graphs

We use the MC-Sym software to build tertiary structure from a base-pairing interaction network (Parisien and Major 2008). MC-Sym is part of the MC-tools package available at: http://www.major.iric.ca/MajorLabEn/MC-Tools.html.

Methods

In this section, we will describe how to extract from a three-dimensional structure, the list of all base pairing interactions. Then, we will describe how to use databases of recurrent motifs to predict the tertiary structure of large RNA molecules.

Classification of base pairing interactions

Watson-Crick (C-G and A-U) and Wobble (G-U) base pairs are the most common type of interactions. They create a scaffold for the tertiary structure. Nonetheless, the analysis of RNA crystal structures revealed a diversity of base pairing interactions that goes far beyond these canonical base pairs. In order to facilitate the description of RNA structures, Leontis and Westhof proposed a complete nomenclature of base pairing interactions (Leontis and Westhof 2001) that aims to provide a better description of the complexity of the tertiary structure. Key to this model is to introduce a detailed representation of the base of nucleotides, which in turn enables us to define a more sophisticated classification of base-pairing interactions. Thereby, a base is abstracted with a right triangle modeling all edges of the molecules that can be potentially involved in an interaction with other nucleotides (See Fig. 1). The three interacting edges are: the Watson-Crick edge, the Hoogsteen edge, and the Sugar edge. The hypotenuse is associated with the Hoogsteen egdge, and the vertex created by the Hoogsteen and Sugar edge represents the root of the base.

Base pairing interactions between nucleotides can now occur between any of these edges. In addition, a complete description of these interactions must also specify the relative orientation of the bases (i.e. the glycosidic bond orientation). Hence, we indicate if the bases are oriented in the same or in opposite directions. These configurations are respectively named *trans* and *cis* configurations.

These parameters enable us to define a complete nomenclature of all possible base-pairing interactions between nucleotides. This catalog is shown in Figure 1 and includes all 12 possible base pairs that are found in RNA structures.

In secondary structure diagrams, base pairs are often represented with specific links. C-G base pairs are represented with two parallel line, A-U base pairs with a single line and G-U base pairs with a circle. Leontis and Westhof also assigned a symbol to each base pairing interaction of their classification. Watson-Crick edges are represented with a circle, Hoogsteen edges with a square and Sugar edges with a triangle. In addition, black symbols represent a *cis* orientation, and white symbols a *trans* orientation. This notation is illustrated at the bottom of each base pair in Figure 1.

Using this nomenclature, RNA tertiary structures can be decomposed into a network of base-pairing interactions. Unlike classical secondary structures, nucleotides are now no longer restricted to interact with at most one other nucleotide. Instead, they can be involved in multiple interactions and create, for instance, base triples. Crossing interactions are also permitted. While this modeling does not obviously encompass all details of three-dimensional atomistic structures, it appears to store most of the information that one need to reconstruct full tertiary structures (Parisien and Major 2008).

Annotation of base-pairing interactions from RNA 3D structures

The rnaview program annotates automatically of all base-pairing interactions found in RNA tertiary structure using the Leontis-Westhof classification (Leontis and Westhof 2001). It can also produce an image of the interaction graph. The command line to run the program is:

```
rnaview –p input.pdb
```

where “input.pdb” is your input three-dimensional structure and the “-p” flag is an optional parameter that indicates to the program to create a visualization of the annotated structure. Figure 2 shows the graphical output of rnaview on the tRNA(Cys) from *Archaeoglobus fulgidus*.

The output files are stored in the same directory as the input file. The list of base pairing interactions is stored in a new text file named after the input file and appended with the suffix “.out”. Similarly, if you used the “-p” option, rnaview also creates a postscript file with a drawing of the base pairing interaction network.

The following code, calculated from the tRNA(Cys) from *Archaeoglobus fulgidus* (2DU3), illustrates a typical output of rnaview:

```
9_12, B:  9 U-G 12 B: S/W tran  syn  n/a
```

```
16_52, B: 16 U-A 52 B: W/H tran     XXIV
18_51, B: 18 G-G 51 B: H/S tran     n/a
18_53, B: 18 G-C 53 B: +/+ cis   syn  XIX
```

The first column gives the indices, separated by an underscore, of the two nucleotides involved in the base pair. These indices are calculated by rnaview and correspond to their positions in the sequence(s) entered in the program. They may eventually differ from the indices stored in the PDB file, which are given in the third and fifth columns.

The letters in the second and sixth columns indicate the chain of the two nucleotides. In our case, the RNA molecule has a single chain named "B" in the PDB file. The types of the nucleotides forming the base pair are indicated in the fourth column.

Finally, the seventh and eighth columns describe the edges involved in the base pair (Watson-Crick, Hoogtseen or Sugar edge), followed by the orientation of their interaction (cis or trans). For instance, in the example above the first row says that "the nucleotide U at index 9, and the nucleotide G at position 12, form a *trans* base pair between the Sugar edge (S) of nucleotide U and Watson-Crick edge (W) of the nucleotide G". It is worth noting that classical Watson-Crick base pairs are annotated with +/+ (C-G base pair) or -/- (A-U base pair). More details on the base pair notation, including "marginal" interactions, can be found in (Yang, et al. 2003) or (Waugh, et al. 2002).

As expected, most base pairs represented in Figure 2 are Watson-Crick base pairs. Nonetheless, we note that non-canonical base pairs tend to get concentrated in specific regions of the secondary structure backbone and create sophisticated local motifs. In fact, this observation is recurrent in most annotated structures. It will give rise to the notion of RNA motifs introduced in the next section.

Finally, it is worth noting that we can alternatively use the program MC-Annotate (Lemieux and Major 2002) to perform similar annotations. However, the classification used by MC-Annotate differs slightly from the one used in the motif databases described further.

RNA motifs

The modeling of RNA tertiary structures into networks of base-pairing interactions revealed recurrent motifs conserved across families of structures. These structural patterns form a base toward better understanding of complex organizations of nucleotides inside hairpins, internal loops and beyond (See previous section for a definition of the secondary structure elements). Several methodologies and databases have been developed to mine and store these data. Among the most popular repositories, we count the RNA 3D Hub maintained by the Bowling Green State University RNA group at http://rna.bgsu.edu/rna3dhub/ (Leontis and Zirbel 2012), and the RNA 3D Motif database developed at the university of Paris-Sud and available at http://rna3dmotif.lri.fr/ (Djelloul and Denise 2008). More recently, international institute for molecular and cell biology in Warsaw introduced a novel motif database RNAbricks, available at http://iimcb.genesilico.pl/rnabricks/.

Beside databases of annotated structures (i.e. the RNA Structure Atlas) and recurrent RNA 3D motifs (RNA 3D Motif Atlas), the RNA 3D Hub offers large suite of online tools and web services. In particular, users can here retrieve local 3D motifs with WebFR3D (http://rna.bgsu.edu/main/webapps/webfr3d/) or align RNA tertiary structures with R3D Align (http://rna.bgsu.edu/main/webapps/webr3dalign/).

Each database developed its own local 3D motif format description, based on the Leontis-Westhof base pair classification. RNA-MoIP use the format introduced with RNA3Dmotif (Djelloul and Denise 2008). An example of an internal loop motif is provided below:

```
Bases: 26_G 27_A 28_G 29_A 30_G 39_U 40_G 41_G 42_U
     ( 39_U )--- C/C - ---( 40_G )
     ( 30_G )--- 5/5 s ---( 40_G )
     ( 30_G )--- +/+ c ---( 39_U )
     ( 40_G )--- C/C - ---( 41_G )
     ( 29_A )--- C/C - ---( 30_G )
     ( 28_G )--- 5/5 s ---( 41_G )
     ( 28_G )--- C/C - ---( 29_A )
     ( 41_G )--- C/C - ---( 42_U )
     ( 26_G )--- +/+ c ---( 42_U )
     ( 27_A )--- C/C - ---( 28_G )
     ( 26_G )--- C/C - ---( 27_A )
```

The first line starting with the key word “Bases” indicates the nucleotides involved in this motif. For each nucleotide, we display its index and its type separated by an underscore. Each following line describes a base pairing interaction in this motif. The syntax of a base pair is structured as follows. On both ends of the row, the two nucleotides (index and type) are shown between parentheses. Then, the type of the interaction is shown in the middle, surrounded by three dashes on each side. The first field indicates the edges involved and the second the base pair orientation (“c” for *cis*, “t” for *trans* and “s” for a stacking). An interaction annotated “C/C” stands for a backbone connection (i.e. consecutive nucleotides) and thus may not be considered as a base pair in our discussion. More information can be found at http://rna3dmotif.lri.fr/help.html.

RNA-MoIP needs to decompose these motifs into components to permit their insertion. Components are largest sequences of contiguous nucleotides that belong to a motif. For instance, in the example above, we have two components ([26_G,27_A,28_G,29_A,30_G] and [39_U,40_G,41_G,42_U]). In fact, the definition of components follows the classical secondary structure loop classification. Hairpins have one single component, bulges and internal loops have two component and k-way junctions have k components. This definition has been introduced with RNA-MoIP to facilitate the description of integer programming equations.

Figure 3 illustrates the definition of motifs. In this example, we identify two motifs (an hairpin and a internal loop) inserted into a single stem. The figure shows the 3D structures and the base pairing interaction graphs of each motif. We observe that

the hairpin has one single component (i.e. one single stranded region), while the internal loop has two.

Prediction of RNA tertiary structures from sequence data

Modeling RNA tertiary structure is the first step toward predicting RNA 3D structures. We will now describe how the knowledge accumulated in motif databases can be used together with RNA secondary structure prediction methods to predict the structure of large RNA molecules from sequence data only. This strategy presented here works in two steps. First, we predict a secondary structure using classical software such as RNAfold (Hofacker, et al. 1994) and refine this prediction by inserting RNA 3D motifs and adding non-canonical base pairs inside the predicted secondary structure with RNA-MoIP (Reinharz, Major and Waldispühl 2012). Next, we use this extended secondary structure to reconstruct three-dimensional models of the molecule using the MC-Sym software (Parisien and Major 2008).

Prediction of a secondary structure scaffold

Our first objective is to create a base-pairing network from sequence data. Since the majority of base pairs in RNA structures are *cis* Watson-Crick base pairs, we first predict a secondary structure (without pseudo-knots) that will be used to build a scaffold of the interaction graph. Secondary structures (without pseudo-knot) can be deterministically predicted with RNAfold, or stochastically generated with RNAsubopt. The command line used to predict the minimum free energy (MFE) secondary structure with RNAfold is:

```
RNAfold --noPS < input.fasta
```

Where input.fasta is a text file (FASTA format recommended) that stores your input sequence. The --noPS flag is not mandatory, but it prevents the program to generate a postscript file drawing the predicted secondary structure. The program returns the input sequence with its MFE secondary structure in bracket format on the line below.

However, as discussed in previous chapters, single energy minimized structures do not always provide the best secondary structure prediction. Instead, it is recommended to perform a deeper exploration of the conformational space and to consider sub-optimal structures (M. Zuker 1989, Ding, Chan and Lawrence 2005). This approach to secondary structure prediction, originally implemented in mfold (Zuker, Mathews and Turner 1999, M. Zuker 1989) and Sfold (Ding and Lawrence 2003), is available with the RNAsubopt program in the Vienna RNA package. The command line for running RNAsubopt is:

```
RNAsubopt -e 3 < input.fasta
```

Where the "-e" option specifies the depth of the sub-optimal search. More specifically, this argument indicates the range (in kCal/mol) from the MFE, within which all sub-optimal structures must be returned. Obviously, the values of that range dependent of the MFE of the sequence, and should be adjusted on a case-by-case basis. In our experiments, a value of 3 kCal/Mol generated 25 secondary

structures; which appears to provide a good representation of the sub-optimal conformational landscape.

The list of sub-optimal secondary structures generated by RNAsubopt (available at http://csb.cs.mcgill.ca/RNAMoIP) provides us a description of the set of potential secondary structure backbones. It will be used as it in the next step.

It is worth noting that other software could have been used to generate the initial secondary structures. RNAstructure (Reuter and Mathews 2010), mfold (Zuker, Mathews and Turner 1999) or Sfold (Ding, Chan and Lawrence 2005) make similar prediction than RNAsubopt. Further, recent software such as MC-Fold (Parisien and Major 2008) and RNAwolf (Höner zu Siederdissen, et al. 2011) have been developed to predict extended secondary structures (including all non-canonical base pairs) directly from sequence data.

Prediction of a complete base pairing interaction graph with motif insertion

We describe how to use RNA-MoIP (Reinharz, Major and Waldispühl 2012) to insert local motifs into secondary structures generated with RNAsubopt. By default, RNA-MoIP aims to inserts motifs from a repository build with RNA3Dmotif (Djelloul and Denise 2008). This repository of non-redundant motifs is included in the distribution of RNA-MoIP and named "No_Redondance_DESC". Nonetheless, advanced users can also either build themselves an up-to-date motif repository using RNA3Dmotif, or use databases available on the RNA 3D Hub (Leontis and Zirbel 2012).

RNA-MoIP is a flexible tool that allows modifications of the secondary structure to permit the insertion of motifs. In particular, the program has the capacity to remove a fixed amount of base pairs from the input secondary structure. This feature is particularly helpful if the predicted secondary structure has incorrectly predicted base pairs that prevent motifs to be inserted.

Lets assume that you work in a directory that contains the RNA-MoIP program (i.e. the python script named "RNAMoIP.py") and that Gurobi has been properly installed. The command line to insert motifs in the sequence and secondary structure with RNA-MoIP is:

```
gurobi.sh RNAMoIP.py -s <sequence> -ss <structure_list> –d
<path_to_repository>
```

Where the argument <sequence> is a string representing the primary structure of the RNA sequence, <structure_list> is the name of the file that stores the secondary structures in bracket format generated by RNAsubopt, and <path_to_repository> is the location of the motif repository. It indicates the path to the motif repository stored in the directory named "CATALOGUE", which is distributed with the RNA-MoIP package at http://csb.cs.mcgill.ca/RNAMoIP/. Assuming that the directory "CATALOGUE" is in your current directory, the value of <path_to_repository> is the string "./CATALOGUE/No_Redondance_DESC/".

RNA-MoIP accepts an additional parameter to control the maximum number of base pairs that can be removed. This parameter can be adjusted with a float number

(between 0 and 1) through the option –r. By default, RNA-MoIP allows up to 30% (i.e. -r 0.3) of base pairs to be removed. This is a reasonable choice as the base pair prediction positive predictive value (PPV) is roughly 60% for classical secondary structure predictors such as RNAfold and mfold (Do, Woods and Batzoglou 2006). Nonetheless, users can decrease this value if they are confident in their predicted secondary structure.

Lets now run a concrete example on the tRNA(Cys) sequence, and lets call "RNAsubopt.out" the file storing the output of RNAsubopt. Then, the RNA-MoIP command line is:

```
gurobi.sh RNAMoIP.py -s
'GCCAGGGUGGCAGAGGGGCUUUGCGGCGGACUGCAGAUCCGCUUUACCCCGGUUCGAA
UCCGGGCCCUGGC'-ss RNAsubopt.out -d
'./CATALOGUE/No_Redondance_DESC/'
```

The program outputs first the usual output of Gurobi (the mathematical solver used by RNA-MoIP). We are interested by the output produced after the line starting with "Best objective". It contains the secondary structure used to insert the motifs, the IDs of the inserted motifs, the positions of the deleted base pairs, and the score of the solution. The RNA-MoIP command above will output the following results:

```
Solution for the secondary structure:
(((((((..((((((...))))))).(((((.......)))))).....(((((.......)))))))))))

Optimal solution nb:  1

Corrected secondary structure:
((((((...(((.........)))..((((.......))))).....(((.........)))).))))))

C-2DU6.D.2-12-22-1
C-3CUL.D.6-51-61-1
C-2DU3.D.3-30-38-1
C-2DU5.D.1-6-10-1
C-2DU5.D.1-24-27-2
C-2DU5.D.1-41-48-3
C-2DU5.D.1-64-66-4
D-15-19
D-14-20
D-13-21
D-26-42
D-52-60
D-7-65

The optimal solutions has as value:
    -663.0
```

The first line starts with "Solution for the secondary structure" and returns the best secondary structure extracted from the list of sub-optimal structure, which has been used to insert to motifs. The next line (starting with "Optimal solution nb") indicates

how many sub-optimal structures have been used. In our case, only one structure can used to build the optimal solution.

The line starting with “Corrected secondary structure” shows the structure used by RNA-MoIP. Since RNA-MoIP may remove some base pairs to insert motifs, this structure can be different to the one returned on the first line.

Then, the program outputs information about the inserted motifs and deleted base pairs. The rows starting with a "D" are the positions of the deleted base pairs. Hence, here the output tells us that the base pairs (15,19), (14,20), (13,21), (26,42), (52,60) and (7,65) have been removed to insert the motifs. We highlight in red the deleted base pairs.

((((((..(((((...)))))).((((.......)))))).....((((.......))))))))))

As indicated in the third line, the secondary predicted (or updated) by RNA-MoIP is:

((((((...(((.........))))..(((((.......))))).....(((((.........)))).))))))

The rows starting with a “C” display information about the inserted motifs. Each row indicates where a component of a motif (i.e. contiguous sequence of a local motif) has been used. The syntax of these lines is:

C-<ID>-<first_index>-<last_index>-<component_number>

Where <ID> is the identifier (or name) of the inserted motif, <first_index> is the first position and <last_index> is the last position of the insertion. The last value <component_number> indicates which component of the motif has been used.

In our example, four motifs have been inserted: Three hairpins and one 4-way junction. For each motif, Table 1 shows the ID, type, and positions at which each component has been inserted. Figure 4 illustrates these results and shows where the motifs and components have been inserted by RNA-MoIP in the secondary structure.

The last line in the output returns the value of the objective function that has been optimized by RNA-MoIP. In our example, the value returned is -663.0. Details about this function can be found in (Reinharz, Major and Waldispühl 2012).

Reconstruction of tertiary structures from base-pairing interaction networks

Now, we describe how to use a base pairing network to build three-dimensional structures with the MC-Sym software (Parisien and Major 2008). This operation requires writing scripts that will be used to run MC-Sym at http://www.major.iric.ca/MC-Sym/.

An MC-Sym script is composed of 6 parts. The first one provides a description of the sequence that is going to be modeled, and the second indicates the set of fragments to be used. The third part is the order in which the fragments will be merged. The fourth and fifth parts describe geometrical constraints to satisfy during the construction of the three-dimensional model. The last segment defines a set of rules for the space exploration. Every line starting by '//' is a comment.

In this example, we show how to model the tRNA(Cys) molecule of *Archaeoglobus fulgidus*, a 71-nucleotides long RNA into which four motifs are inserted by RNAMoIP (See previous section). Each part of the script is described and explained separately.

Part 1: Sequence Definition

The first step defines the sequence we are modeling. The syntax is:

```
// ==================== Sequence ====================
sequence( r A1  GCCAGGGUGGCAGAGGGGCUUUGCGGCGGACUGCAGAUCCGCUUUACCCCGGUUCGAAUCCGGGCCCUGGC )
//              ((((((...((((((...))))))..((((.......)))).....(((((((...)))))))).))))))
//              12345678901234567890123456789012345678901234567890123456789012345678901
//                        1         2         3         4         5         6         7
```

The keyword “sequence” states that we are defining a sequence. The program requires three arguments that are indicated between the parentheses. First, 'r' specifies that we are working with an RNA sequence. Next, 'A1' sets as identifier for the sequence 'A' and defines the first position as '1'. Finally, we end by the sequence and close the parenthesis. The two lines of comment that follows are just here to improve readability.

Part 2: Fragment Definition

This is probably the most important and delicate step of the script. In MC-Sym, the basic units are called cycles. The second step aims to define which cycles and motifs will be used to build the structure. Cycles, motifs and other structural elementary blocks used by MC-Sym are called fragments. The order in which they entered is not important, but we must ensure that every position is included in a fragment and that all fragments overlap.

In the following example, we illustrate how to insert a basic cycle (i.e. not a motif). Here, the cycle is a stack between two consecutive base pairs. This is the most common usage of cycles as most Watson-Crick and Wobble base pairs are predicted by secondary structure predictors and thus not covered by RNA motifs. The code below indicates how and where to map this cycle. In this case, a stacked base pairs at positions 1, 2, 70 and 71.

```
ncm_01 = library(
    pdb( "MCSYM-DB/2_2/GCGC/*R20*.pdb.gz" )
    #1:#2, #3:#4 <- A1:A2, A70:A71
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )
```

We start with a unique identifier (i.e. "ncm_01"), followed by the keyword "library". It indicates to MC-Sym where to retrieve the basic fragments.

On the second line, we provide specifications of the files within that library. Here, our library is composed of PDB files, followed by the path "MCSYM-DB/2_2/GCGC/*_1.pdb.gz". MCSYM-DB is the location of all standard library fragments. MC-Sym fragments are sorted according to the number of consecutive nucleotides in each component. Here, we have two times two consecutive nucleotides (1-2 and 70-71). Therefore, we must look into the subfolder “2_2”. Next, we must explicitly tell which sequence of nucleotides are involved, in the order 5' to 3'. Since the nucleotides at positions 1, 2, 70 and 71 are respectively 'G', 'C', 'G', 'C',

the next path is “GCGC”. We end by “*.pdb.gz” to consider all possible structures for this specific fragment. Specific subsets of those fragments can be chosen as specified in the MC-Sym documentation.

On the third line, we provide detailed information on the positions onto which the fragment will be mapped. The syntax is composed of two parts, separated by the symbol "<-". The left-hand side indicates the segments associated with each consecutive component. Segments are represented with an interval, and nucleotides are numbered sequentially according to their position in the motif. The right-hand side defines the specific positions (still as intervals or segments) onto which these components/segments must match. In our sample code, we have two stretches of two nucleotides. We write it “#1:#2, #3:#4”, which also implies that the first part is composed of the two first nucleotides and the second one of the next two. On the right hand side, “A1:A2, A70:A71” describes which sequence must be mapped. The nucleotides at from position 1 to 2 for the first component, and the nucleotides at from position 70 to 71 for the second component. “A” is a unique identifier of the sequence to be used, as defined in the first part. The last line is a set of constraints for this specific fragment. Since this information is non-essential here, we kindly redirect the reader to the MC-Sym documentation for more details. It is also important to not forget to close the parenthesis opened after the keyword "library".

Now, we must specify how to declare the fragments inserted by RNAMoIP. The simplest one is a hairpin. Lets start with the hairpin “1DU6.D.2”, between positions “12” and “22”. The code is:

```
pdbs_hairpin_1 = library(
    pdb ("/u/reinharz/RNAMOIP-DB-VIEW3D/2DU6.D.2/*.pdb")
    #1:#11 <- A12:A22
    )
```

The syntax is very similar to those of the previous example. The first line is the same as for the stacked base pair: a unique identifier followed by the keyword “library”. However, the path to the library is different. All RNA-MoIP motifs are contained in a repository that is accessible with the path "/u/reinharz/RNAMOIP-DB-VIEW3D/<Motif Identifier>". The path must always end by "*.pdb" to indicate that we want to consider all possible structures in the folder. On the third line, we define the positions involved. In this case, the motif contains 11 consecutive nucleotides, thus the left hand side is "#1:#11". Those nucleotides are at positions "12" to "22" in our sequence, defining the right hand side. We end by closing the parenthesis of "library".

Finally, motifs with multiple stretches of consecutive nucleotides are a slightly more complicated to specify. We illustrate that case with the 4-way junction "2DU5.D.1". Using the same syntax as above, we write the code as follows.

```
pdbs_4_way = library(
    pdb ("/u/reinharz/RNAMOIP-DB-VIEW3D/2DU5.D.1/*.pdb")
    #1:#5, #6:#9, #10:#17, #18:#20 <- A6:A10, A24:A27, A41:A48, A64:A66
    )
```

In this case, our motif is composed of 4 components, between positions "6" and "10", "24" and "27", "41" and "48" and "64" and "66" in the input sequence. Each component of consecutive nucleotides needs to be described individually.

Part 3: Merging Fragments

Once all fragments are specified, we must define in which order they are going to be assembled together. One fragment must be selected to start the construction. All others will be sequentially added; with the constraint that each new fragment must overlaps with any of the previous one. The syntax to assemble the first two stacked base pairs, uses the unique names used in the fragments definitions (i.e. ncm_01 and ncm_02), and is written as follows.

```
structure = backtrack
    (
        ncm_01
        merge( ncm_02  1.5 )
    )
```

It is important to always start with the keywords “structure = backtrack”. The assembly is indicated between two parentheses. The starting point of our model is “ncm_01” and must be written on the first line. On the second line we use the keyword “merge” followed, between parentheses, by an arguments that indicates the next fragment, and another argument that specifies an “error” threshold that can be tolerated by the program to extend the structure. By default we use a value of “1.5”. More information detail on this parameter can be found in the MC-Sym documentation.

Part 4 & 5: Constraints

MC-Sym requires several general parameters and constraints to perform the three-dimensional re-construction. A standard set of values is provided in the following code and can be used as it is.

```
// ================== Backtrack Restraints ==================
clash
(
    structure
    1.5 !( pse || lp || hydrogen )
)
backtrack_rst
(
    structure
    width_limit  = 25%,
    height_limit = 33%,
    method       = probabilistic
)
// ================== Ribose Restraints ==================
ribose_rst
(
    structure
    method    = ccm,
    pucker    = C3p_endo,
    threshold = 2.0
)
```

Part 6: Space Exploration Parameters

Finally, the last step is to determine the boundaries of the space to explore. We also specify the time limit and the maximal number of structures to output. MC-Sym will stop as soon as one of those conditions is fulfilled. The other parameter options are standard and can be found in the documentation of MC-Sym.

```
// ================== Exploration Initialization =========
explore
(
    structure
    option(
        model_limit = 1000,
        time_limit  = 30m,
        seed        = 3210 )
    rmsd( 3.0 sidechain && !( pse || lp || hydrogen ) )
    pdb( "structure" zipped )
)
```

Where, the key word "model_limit" specifies the maximal number of structure to generate, and "time_limit" the maximal amount of time allowed to run MC-Sym. Here, we use an upper bound of 30 minutes with a maximum of 1000 structures. The time needed to generate structures can vary. In our benchmark (Reinharz, Major and Waldispühl 2012), we have seen that about half an hour of computation was often enough to produce solutions with molecules with sizes up to one hundred nucleotides. Within this time frame, the maximal number of structures created will rarely reach one thousand in that time. Importantly, motifs with large number of components will highly restrict the exploration of the space, thus the time needed to explore the conformational space. In absence of any motif with three or more components, increasing the time limit to two days is a reasonable step. In that case, the number of structures could increase to five or even ten thousands, and we can expect much more diversity in the sample set.

It is difficult to determine in advance how many structure samples are necessary to provide a good representation of the conformational landscape. Numbers between 100 and 1000 are good starts, but more might be occasionally needed. We acknowledge that large numbers of structures are difficult to analyze. Fortunately, as we will see in the next section, the MC-Sym server offers several tools to facilitate these tasks.

The key word "seed" is a random number used to initialize MC-Sym. Its value has little importance in the context of this discussion, and users can modify it if they wish.

The complete source code of this MC-Sym script is available at http://csb.cs.mcgill.ca/RNAMoIP/ .

Retrieving, optimizing, analyzing and visualizing structures

Once submitted, the script will run on MC-Sym servers and the URL of a result page will be returned to the user. This page offers multiple options to optimize, analyze and retrieve the results. To access these options, the user must access the web page "commands.html" located in the results directory.

The energy minimization of the MC-Sym structure is probably one of the most important options. We recommend any user to run it before analyzing or visualizing the results. The “steepest descent” option returns satisfactory results in a short time, but more sophisticated and slower techniques (E.g. simulated annealing) are also available.

Clustering of structures using the k-means algorithm is another useful option that enables the user to automatically identify the most significant structures in the set or structures returned by MC-Sym.

A PDB model of all predicted structures is available at the root of the directory. Each model can be visualized with molecular viewer such as PyMOL or Jmol. Figure 5 show an example of a structure predicted with our RNA-MoIP and MC-Sym pipeline, aligned with the experimental structure (Fukunaga and Yokoyama 2007).

Notes

The motifs database used by RNA-MoIP includes motifs from the PDB model 2DU3 (tRNA(Cys) from *Archaeoglobus fulgidus*) database. In applications on other (new) molecules, it will be very unlikely to have motifs from the same molecule in the motif repository. Fortunately, even in the absence of motifs, the benchmark performed in (Reinharz, Major and Waldispühl 2012) shows that the quality of the structure prediction remains high.

Bibliography

Šponer, J., M. Otyepka, P. Banáš, K. Réblová, and N. Walter. "Molecular Dynamics Simulations of RNA Molecules." In *Innovations in Biomolecular Modeling and Simulations*, by Tamar Schlick, 129-155. Cambridge: The Royal Society of Chemistry, 2012.

Bekaert, M., et al. "Towards a computational model for -1 eukaryotic frameshifting sites." *Bioinformatics* 19 (2003): 327-335.

Berman, H.M., et al. "The Nucleic Acid Database: A Comprehensive Relational Database of Three-Dimensional Structures of Nucleic Acids." *Biophysical Journal* 63 (1992): 751-759.

Bernauer, J., X. Huang, A.Y. Sim, and M. Levitt. "Fully differentiable coarse-grained and all-atom knowledge-based potentials for RNA structure evaluation." *RNA* 17, no. 6 (2011): 1066-1075.

Bernstein, F.C., et al. "The Protein Data Bank: A Computer-based Archival File For Macromolecular Structures." *Journal of Molecular Biology* 112 (1977): 535.

Das, R., J. Karanicolas, and David Baker. "Atomic accuracy in predicting and designing noncanonical RNA structure." *Nature Methods* 7 (2010): 291-294.

Das, Ruiji, and David Baker. "Automated de novo prediction of native-like RNA tertiary structures." *PNAS* 104, no. 37 (2007): 14664-14669.

Ding, F., S. Sharma, P. Chalasani, V.V. Demidov, N.E. Broude, and N.V. Dokholyan. "Large scale simulations of 3D RNA folding by discrete molecular dynamics: From structure prediction to folding mechanisms." *RNA* 14 (2008): 1164-1173.

Ding, Y, CY Chan, and CE Lawrence. "RNA secondary structure prediction by centroids in a Boltzmann weighted ensemble." *RNA* 11 (2005): 1157-1166.

Ding, Y., and C.E. Lawrence. "A statistical sampling algorithm for RNA secondary structure prediction." *Nucleic Acids Research* 31 (2003): 7280-7301.
Djelloul, M., and A. Denise. "Automated motif extraction and classification in RNA tertiary structures." *RNA* 14 (2008): 2489-2497.
Do, C.B., D.A. Woods, and S. Batzoglou. "CONTRAfold: RNA secondary structure prediction without physics-based models." *Bioinformatics* 22, no. 14 (2006): e90-e98.
Fukunaga, R., and S. Yokoyama. "Structural insights into the first step of RNA-dependent cysteine biosynthesis in archaea." *Nature Structural Molecular Biology* 14 (2007): 272-279.
Höner zu Siederdissen, C., S.H. Bernhart, P.F. Stadler, and I.L. Hofacker. "A Folding Algorithm for Extended RNA Secondary Structures." *Bioinformatics* 27, no. 13 (2011): i129-i136.
Hainzl, T., S. Huang, and A.E. Sauer-Eriksson. "Structure of the SRP19 RNA complex and implications for signal recognition particle assembly." *Nature* 417 (2002): 767-771.
Hofacker, I.L., W. Fontana, P.F. Stadler, S. Bonhoeffer, M. Tacker, and P. Schuster. "Fast Folding and Comparison of RNA Secondary Structures." *Monatshefte f. Chemie* 125 (1994): 167-188.
Jonikas, M.A., et al. "Coarse-grained modeling of large RNA molecules with knowledge-based potentials and structural filters." *RNA* 15, no. 2 (2009): 189-199.
Lemieux, S., and F. Major. "RNA canonical and non-canonical base pairing types: a recognition method and complete repertoire." *Nucleic Acids Research* 30, no. 19 (2002): 4250-4263.
Leontis, N.B., and C.L. Zirbel. *Nonredundant 3D Structure Datasets for RNA Knowledge Extraction and Benchmarking.* Vol. 27, in *RNA 3D Structure Analysis and Prediction*, edited by N.B. Leontis and E. Westhof, 281-298. Springer Berlin Heidelberg, 2012.
Leontis, Neocles B., and Eric Westhof. "Geometric nomenclature and classification of RNA base pairs." *RNA* 7 (2001): 499-512.
Lorenz, R., et al. "ViennaRNA Package 2.0." *Algorithm for Molecular Biology* 6, no. 26 (2011).
Martinez, H.M., J.V. Jr. Maizel, and B.A. Shapiro. "RNA2D3D: a program for generating, viewing, and comparing 3-dimensional models of RNA." *Journal of Biomolecular Structure & Dynamics* 25, no. 6 (2008): 669-683.
Nikulin, A., et al. "Crystal structure of the S15-rRNA complex." *Nature Structural Biology* 7 (2000): 273-277.
Parisien, M., and F. Major. "The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data." *Nature* 452 (2008): 51-55.
Poursina, M., K.D. Bhalerao, S.C. Flores, Anderson K.S., and Laederach A. "Strategies for articulated multibody-based adaptive coarse grain simulation of RNA." *Methods Enzymol.* 487 (2011): 73-98.
Reinharz, V., F. Major, and J. Waldispühl. "Toward 3D structure prediction of large RNA molecules: An integer programming framework to insert local 3D motifs in RNA secondary structure." *Bioinformatics* 28, no. 12 (2012): i207-i214.

Reuter, J. S., and D. H. Mathews. "RNAstructure: software for RNA secondary structure prediction and analysis." *BMC Bioinformatics* 11 (2010): 129.
Rother, M., K. Rother, T. Puton, and J.M. Bujnicki. "ModeRNA: a tool for comparative modeling of RNA 3D structure." *Nucleic Acids Research* 39, no. 10 (2011): 4007-4022.
Szewczak, A.A., P.B. Moore, Y.L. Chang, and I.G. Wool. "The conformation of the sarcin/ricin loop from 28S ribosomal RNA." *Proceedings of the National Academy of Sciences* 90, no. 20 (1993): 9581-9585.
Vitreschak, A.G., D.A. Rodionov, A.A. Mironov, and M.S. Gelfand. "Riboswitches: the oldest mechanism for the regulation of gene expression?" *Trends in Genetics* 201, no. 1 (2004): 44-50.
Wang, Z., and J. Xu. "A Conditional Random Fields Method for RNA Sequence-Structure Relationship Modeling and Conformation Sampling." *Bioinformatics*, 2011: i102-i110.
Waugh, A., et al. "RNAML: a standard syntax for exchanging RNA information." *RNA* 8, no. 6 (2002): 707-717.
Yang, H., et al. "Tools for the automatic identification and classification of RNA base pairs." *Nucleic Acids Research* 31, no. 13 (2003): 3450-3460.
Zhao, Y., Y. Huang, Z. Gong, Y. Wang, J. Man, and Y. Xiao. "Automated and fast building of three-dimensional RNA structures." *Scientific Reports*, 2012.
Zuker, M., D.H. Mathews, and D.H. Turner. "Algorithms and Thermodynamics for RNA Secondary Structure Prediction: A Practical Guide." Edited by J. Barciszewski and B.F.C. Clark. *NATO ASI Series.* Poznan: Kluwer Academic Publishers, 1999.
Zuker, Michael. "On finding all suboptimal foldings of an RNA molecule." *Science* 244, no. 4900 (1989): 48-52.

Figure Legend

Figure 1: Base modeling and Leontis-Westhof base pair classification. The base of a nucleotide is represented with a right triangle. The hypotenuse represents the Hoogsteen edge (noted “H”), and the other sides are associated with the Watson-Crick edge (noted “W”) and Sugar edge (noted “S”). This figure represents all 12 base pair interactions with *cis* or *trans* orientation.

Figure 2: rnaview annotation of a crystal structure of tRNA(Cys) from *Archaeoglobus fulgidus* (Fukunaga and Yokoyama 2007).

Figure 3: Example of 3D RNA motif insertions in a secondary structure. In green, we show the position of the hairpin motif “1F7Y.B.6”, and in blue we indicate the position of the internal loop motif “1FKA.A.51”. On the left side of the motif IDs, we display a 3D structure of the motif together with its base pairing interaction graph.

Figure 4: Location of RNA motifs inserted by RNA-MoIP in the secondary structure of tRNA(Cys) from *Archaeoglobus fulgidus*. Three hairpins (2DU6.D.2 in green, 2CUL.D.6 in magenta, and 2DU3.D.3 in blue) and one 4-way junction (2DU5.D.1 in red) have been inserted. We note the 4 components of the 4-way junction, which correspond to the 4 single strand of the multi-loop.

Figure 5: 3D structure of the tRNA(Cys) from *Archaeoglobus fulgidus* predicted with the RNA-MoIP+MC-Sym pipeline (in blue), aligned with its crystallographic model (Fukunaga and Yokoyama 2007).

Table 1: Description of motifs inserted by RNA-MoIP in the sub-optimal secondary structures generated by RNAsubopt for the tRNA(Cys) from *Archaeoglobus fulgidus*.

Motif ID	Type	Indices of insertion sites
2DU6.D.2	hairpin	[12,22]
3CUL.D.6	hairpin	[51,61]
2DU3.D.3	hairpin	[30,38]
2DU5.D.1	4-way junction	[6,10],[24, 27],[41,48],[64,66]

Table 1

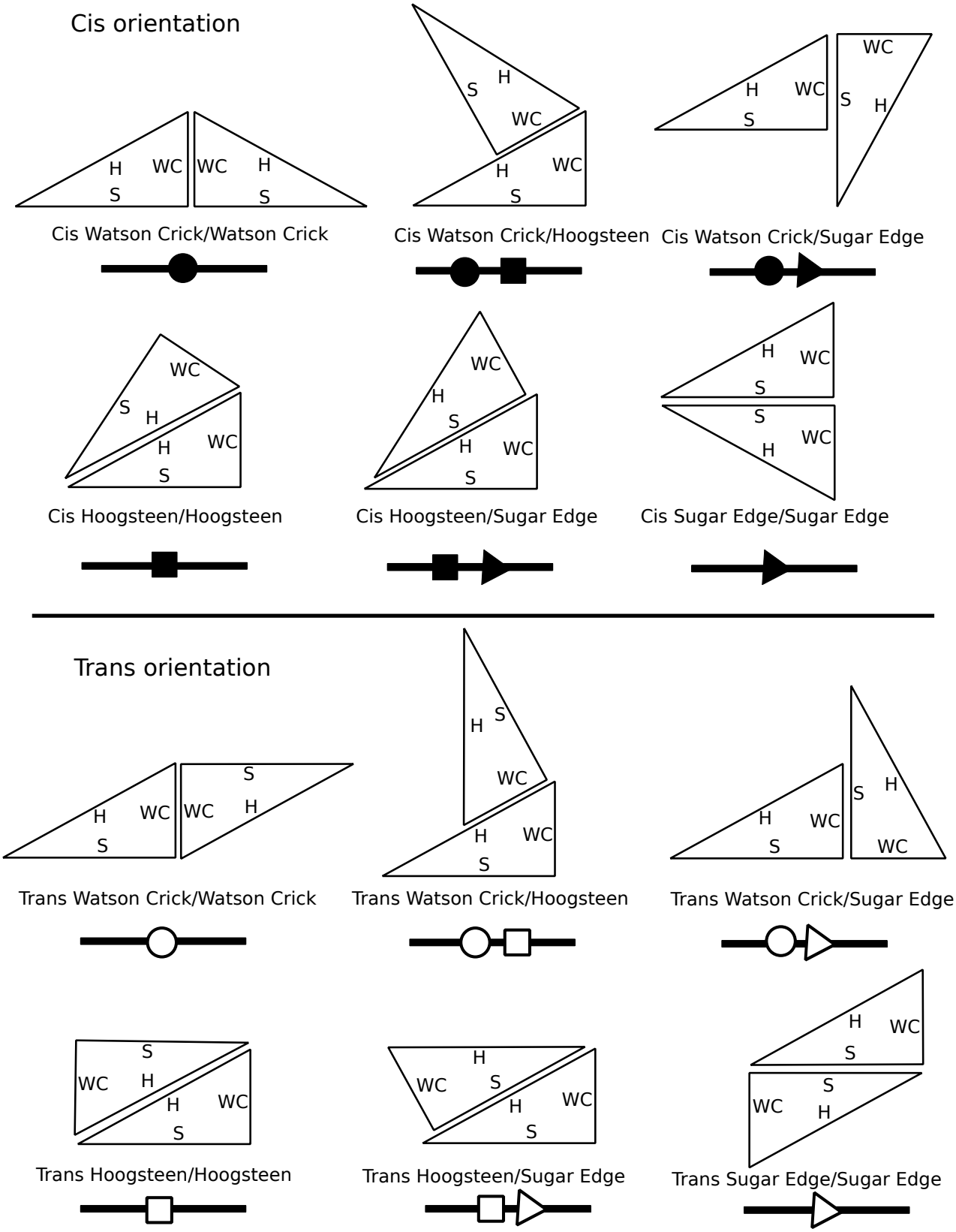
Cis orientation
H
WC
S
WC
H
S
Cis Watson Crick/Watson Crick
S
H
WC
H
WC
S
Cis Watson Crick/Hoogsteen
H
WC
S
WC
S
H
Cis Watson Crick/Sugar Edge
WC
S
H
H
WC
S
Cis Hoogsteen/Hoogsteen
WC
H
S
H
WC
S
Cis Hoogsteen/Sugar Edge
H
WC
S
S
WC
H
Cis Sugar Edge/Sugar Edge
Trans orientation
H
WC
S
S
WC
H
Trans Watson Crick/Watson Crick
H
S
WC
H
WC
S
Trans Watson Crick/Hoogsteen
H
WC
S
S
H
WC
Trans Watson Crick/Sugar Edge
S
WC
H
H
WC
S
Trans Hoogsteen/Hoogsteen
H
WC
S
H
WC
S
Trans Hoogsteen/Sugar Edge
H
WC
S
S
WC
H
Trans Sugar Edge/Sugar Edge

Figure 1

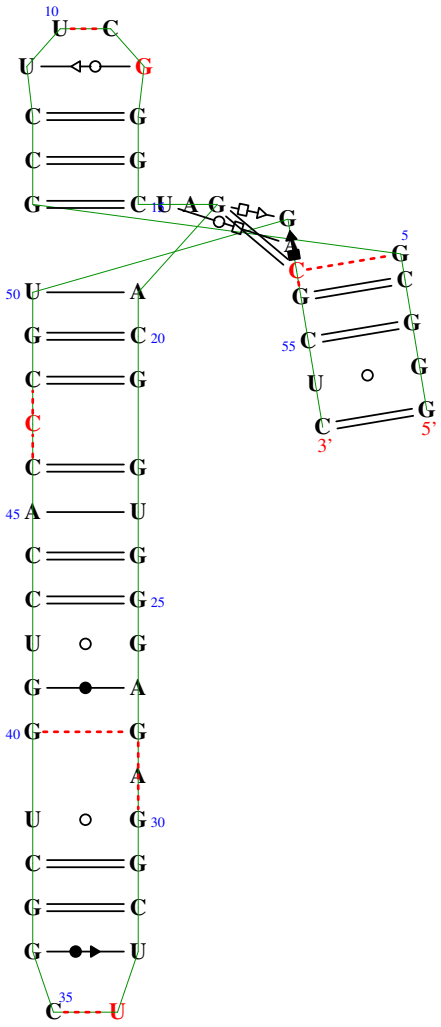
10
U
C
U
G
C
G
C
G
G
C
U
A
G
G
5
G
C
C
G
C
G
C
G
U
G
C
3'
5'
50
U
A
G
C
20
C
G
C
C
G
C
G
45
A
U
C
G
C
G
25
U
G
G
A
40
G
G
A
U
G
30
C
G
G
C
G
U
35
C
U

Figure 2

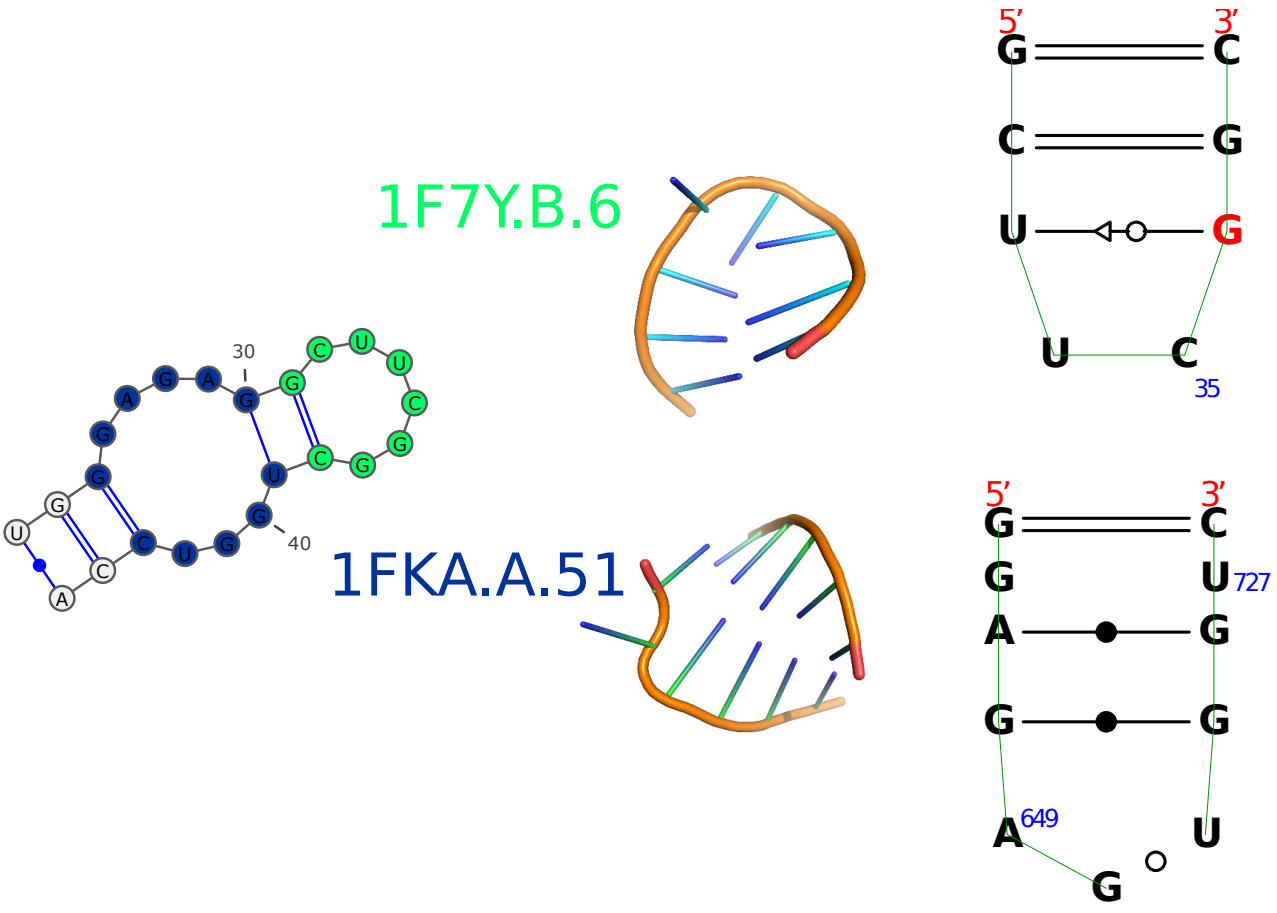

Figure 3

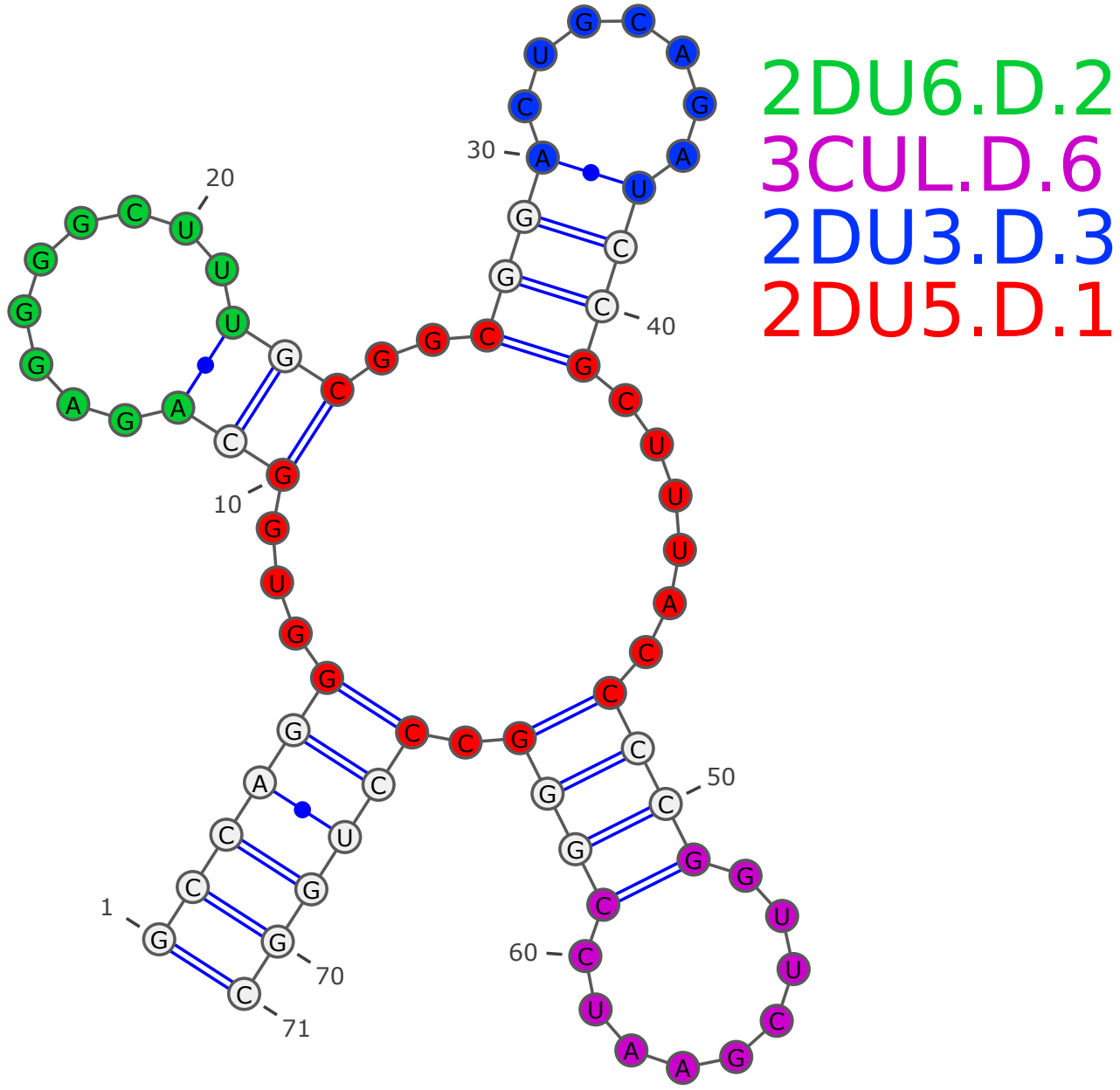

Figure 4

Figure 5