Towards 3D structure prediction of large RNA molecules:
an integer programming framework to insert local 3D motifs in RNA secondary structure

Vladimir Reinharz

Master of Science

Computer Science

McGill University

Montreal, Quebec

15-08-2012

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Master of Science

ABSTRACT

The prediction of RNA three-dimensional structures from its sequence only is a milestone to RNA function analysis and prediction. In recent years, many methods addressed this challenge, ranging from cycle decomposition and fragment assembly to molecular dynamics simulations. However, their predictions remain fragile and limited to small RNAs. In this work, we introduce `RNA-MoIP`, a new framework incorporating the novel local motif information available in databases for the prediction of RNA structures. We show that our approach (i) improves the accuracy of canonical base pair predictions, (ii) identifies the best secondary structures in a pool of sub-optimal structures, and (iii) predicts accurate 3D structures of large RNA molecules.

ABRÉGÉ

Un objectif principal de l'analyse fonctionnelle et prédictive de l'ARN est d'obtenir sa structure tridimensionnel à partir de sa séquence. Pour résoudre ce problème, plusieurs méthodes ont été développées durant les dernières années, telles la décomposition cyclique, l'assemblage de fragments et la simulation de dynamiques moléculaire. Cependant, leurs capacacités prédictives restent limitées. Nous avons mis au point un nouvel outil, `RNA-MoIP`, permettant d'incorporer l'information des motifs locaux nouvellement accessibles dans des bases de données pour la prédiction de structures d'ARN. Nous montrons que notre approche (i) améliore la prédiction des paire de bases canoniques (ii) identifie la meilleure structure secondaire dans un ensemble de sous-optimaux et (iii) prédit des structures 3D précises pour de large molécules d'ARN.

ACKNOWLEDGEMENTS

I thank my supervisor Dr.Jérôme Waldispühl for his guidance and support before and through my Master's studies. I would also like to thank Prof. François Major, Marc-Frédérick Blanchet and Karine Saint-Onge for their help with the `MC-Sym` program, as well as Yann Ponty and Mohit Singh for their useful comments and suggestions when designing the IP framework.

CONTRIBUTION OF AUTHORS

Vladimir Reinharz has performed the research described in this thesis. He designed the method and experiments, implemented the full program and performed the experiments. He wrote the paper [19] in collaboration with Jérôme Waldispühl. François Major and Jérôme Waldispühl provided guidance.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
Introduction

Ribonucleic acids (RNAs) are molecules performing a broad range of functions in cells. Many examples have been found of RNAs serving to catalyze chemical reactions, such as the RNase P or the group II introns. Other groups, such as microRNAs, regulate gene expression by hybridization to messenger RNA. To perform these vast array of functions, RNAs need to fold into specific three-dimensional structures that are directly encoded in their nucleotide sequence. The structural information is therefore essential to gain information about the function. The prediction of RNA three-dimensional structures from its sequence only is thus a milestone to RNA function analysis and prediction. Nonetheless, experimental determination of RNA structures remains time-consuming and technically challenging. Therefore, it needs fast and reliable computational tools to help predict them.

1.1 RNA structure

RNA molecules are ordered sequences of nucleotides. Each nucleotide is composed of a ribose sugar and a phosphate group, linking the nucleotides together and forming the backbone of the strand. It also contains one of the four nucleobases: **A**denine, **U**racile, **G**uanine or **C**ytosine.

Those molecules fold into complex 3D structures stabilized by interaction between the nucleotides. Any of the nucleotides can pair with any other in 12 different

basic geometric configurations [14]. This quickly rises to an intractable number of possibilities.

Historically, a lot of disparate information about RNAs energy and structure could be found until the Salser review in 1978 [21]. The energy for only 3 types of interactions, the canonical base pairs (i.e. **A**-**U**, **G**-**C** and **G**-**U**), were strong enough to have been experimentally evaluated.

Thus, mathematical models focused on predicting structures using as basic pieces the canonical base pairs. The structure, when restrained to the canonical base pairs, is called the secondary structure. Formally a secondary structure can be defined as follows.

Given a sequence of n nucleotides $s := s_1 s_2 \cdots s_n, s_i \in \{\mathbf{A}, \mathbf{U}, \mathbf{G}, \mathbf{C}\}$, a `secondary structure` S over s is a set of ordered pairs corresponding to base pair positions, which satisfies the following requirements.

1. Only Watson-Crick or GU wobble pairs allowed: If $(i, j) \in S$, then $i < j$ and (s_i, s_j) must be one of the following canonical base pairs: $\{(\mathbf{A}, \mathbf{U}), (\mathbf{U}, \mathbf{A}), (\mathbf{G}, \mathbf{C}), (\mathbf{C}, \mathbf{G}), (\mathbf{G}, \mathbf{U}), (\mathbf{U}, \mathbf{G})\}$.
2. Minimal base pairing distance: If $(i, j) \in S$, then $j - i > \theta$.
3. No pseudo-knots: If (i, j) and $(k, l) \in S$, then $j < k$ or $l < i$.
4. Only one interaction per nucleotide: If (i, j) and $(i, k) \in S$, then $j = k$; if (i, j) and $(k, j) \in S$, then $i = k$.

The minimal base pairing distance θ for hairpins is a physical constraint usually set around 3.

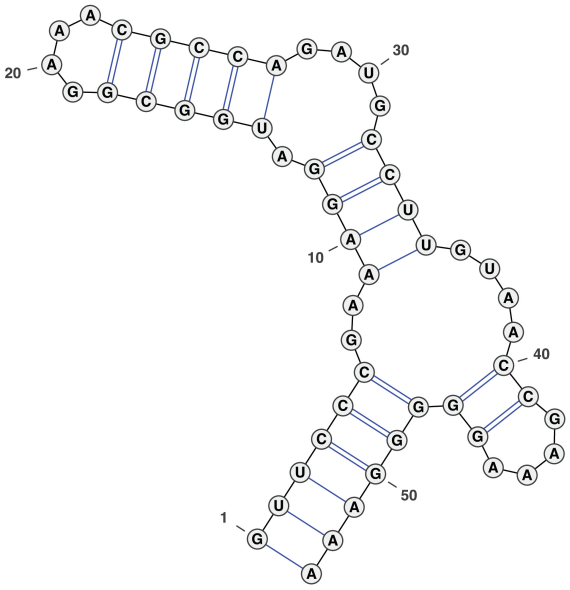


Figure 1–1: Representation of the secondary structure of an RNA as a graph and as a well balanced parenthesized equation.

We show in Fig. 1–1 two representations of a secondary structure, as a graph and as a well balanced parenthesized equation. The fact that the structures disallow pseudo-knots allows the writing of the structure as a well balanced parenthesized equation with one type of parenthesis.

This led, in 1981, to the highly successful dynamic programming algorithm of Zuker and Stiegler [30] for finding the minimal free energy secondary structure of an RNA sequence, given those experimental values.

Due to the strength of their bonds, the base pairs considered in the secondary structure form a scaffold for the entire 3D structure. Nonetheless, the rigidity of those interaction impedes them to create the complex forms allowing molecules to hold

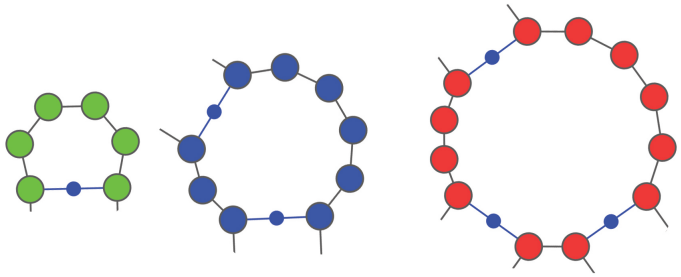


Figure 1–2: An hairpin, interior loop and 3-way junction as represented in a secondary structure.

their functions. Thus, predicting efficiently secondary structures is a first step to the daunting task of obtaining an all atoms 3D model but does not contain information for regions void of the canonical base pairs (i.e. **A**-**U**, **G**-**C** and **G**-**U**). Those regions are the ones folding into the most complex shapes and therefore containing most valuable information. In secondary structures, there are 3 basic shapes containing no base pairs: hairpins, interior loops and k-way junctions (presented in Fig. 1–2).

As shown in [25], roughly 50% of interactions are in fact canonical base pairs. The pieces of the secondary structures void of canonical base pairs are those containing the remaining ones, explaining the diversity of shapes observed. We present in Fig. 2–1 an annotated secondary structure representation of an RNA. The blue lines represent nucleotides canonical interactions. We show in green two hairpins, in blue one interior loop and in red a 3-way junction. Also presented in Fig. 2–1 are the 3D structures of those 4 pieces as stored in the Protein Data Bank [1] (`www.pdb.org`). We can notice that they contain a high level of organization.

1.2 RNA structure prediction

During the last few years, several groups have developed fully automated RNA three-dimensional structure prediction programs. However, those three-dimensional structure predictors have limitations. Currently, their time requirement and/or their accuracy restrict their application range to sequences with less than 50 nucleotides.

In contrast, restraining to the secondary structure, classical predictors, such as `RNAstructure` [20], `RNAfold` [8], `unafold` [15], `contrafold` [5] or `contextfold` [28], are fast and reliable on sequences with more than 100 nucleotides. `MC-Fold` [17] and `RNAwolf` [29] expanded these techniques to predict extended secondary structures (i.e. including non-canonical interactions), but these programs remain limited to predict nested secondary structures without k-way junctions, thus, precisely lacking the structural motifs shaping the RNA 3D structure.

Thus, *ab-initio* 3D structure prediction of large RNA molecules (i.e. more than 50 nucleotides in our context) is still an open question. To overcome this barrier, new models are required. However, due to the paucity of structural data available, the design of a complete model accounting for all the subtle three-dimensional structural variations observed in experimentally determined structures is unlikely.

1.3 Related Work

The decomposition of RNA structures in elementary blocks was first introduced by Lemieux and Major [13] who proposed a description of RNA secondary structures (including non-canonical interactions) based on cycles. More recently, the analysis of experimental 3D structures revealed that similar 3D motifs can be found in multiple

unrelated structures. Here, we define a motif as a group of nucleotides that adopt a specific 3D shape and interaction pattern (including non-canonical interactions). Several groups have developed computational methods to extract and classify RNA 3D motifs. Those algorithms have been run on all structures in the *PDB* [1] and have been consolidated in databases. The most popular databases are `FR3D` [22], `Rna3Dmotif` [4], and `RNAjunction` [2]. Importantly, these databases identify 3D motifs involving three or more segments of the same molecule defined as k-way junctions (when the motif is the branching point of several helical segments). Such motifs are important because they are precisely those shaping the 3D structure of an RNA molecule.

Despite the knowledge accumulated in these databases, the integration of this information into current models remains complicated. First, the classification of RNA motifs can be ambiguous (i.e. a motif and its sub-motifs can match different database entries). Next, the structural compatibility between two or more motifs can be difficult to resolve (i.e. how to concatenate two motifs). It is worth noting that a method to predict the topological family of a given three-way junction has been recently introduced by [12].

Interestingly, to complement the secondary structure programs, [16] and [10] implemented semi-automated methods (resp. `RNA2D3D` and `assemble`) for building three-dimensional models from known/predicted secondary structure information. These programs provide intuitive interfaces enabling their users to insert three-dimensional motifs and modify backbone angles of a coarse grained input structure.

From this standpoint, the hierarchical approaches (i.e. `RNA2D3D` and `assemble`) appear well suited to the prediction of large RNA structures. Their advantage resides in their capacity to benefit from the high accuracy of classical secondary structure predictors (i.e. thermodynamic or comparative models) to build a scaffold of the structure, and then to leave to the user the task of decorating the model with the various structural motifs found in databases. Although this strategy is flexible, it is time-consuming and requires human participation. Recently, [3] developed `RMDetect`, a method to predict G-bulge loops, kink-turns, C-loops and tandem-GA loops in RNA secondary structures. But the prediction of more complex motifs such as the k-way junctions and the construction of 3D RNA structures remain open problems.

1.4 Our Contribution

The methods developed in this thesis are based on a recent idea suggesting that RNA 3D structures share common structural subunits. We introduce `RNA-MoIP`, an integer programming (IP) framework for inserting RNA 3D motifs inside known (or predicted) RNA secondary structures. Our method refines predicted secondary structures (i.e. removes incorrect canonical base-pairs) to accommodate the insertion of RNA 3D motifs (i.e. hairpins, internal loops and k-way junctions). Integer programming techniques have gained a lot of interest recently as they provided state-of-the-art methods for predicting RNA secondary structures with pseudo-knots [18, 23]. One of their strengths resides in their flexibility and capacity to incorporate heterogeneous

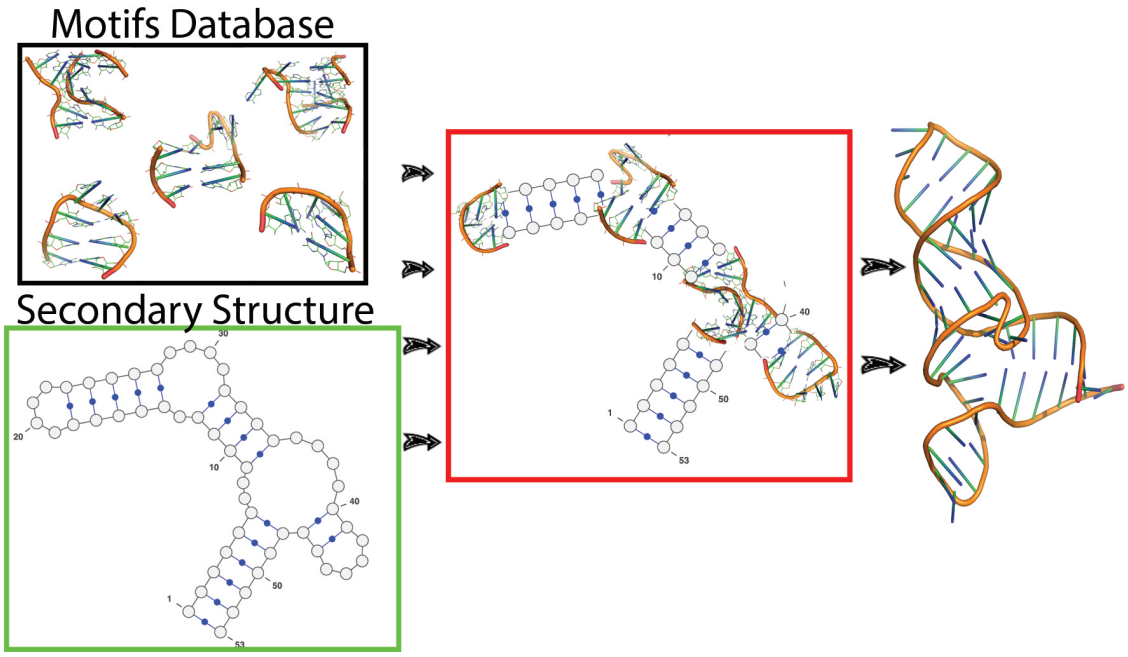


Figure 1–3: The RNA-MoIP workflow.

The motifs database and the secondary structure are the inputs to RNA-MoIP which find the motifs fitting the best, under the objective function inside the secondary structure. Then MC-Sym is used to recreate an all atoms model of the structure.

constraints, a valuable advantage when it comes to incorporate k-way junctions. A schematic of the workflow is presented in Fig. 1–3.

We use our predictions as a template to generate putative RNA 3D structures using the MC-Sym [17] software. We benchmarked RNA-MoIP on a set of 9 RNAs with sizes varying from 53 to 128 nucleotides. We show that our approach improves the accuracy of canonical base pair predictions, identifies the best secondary structures

in a pool of sub-optimal structures, and predicts accurate 3D structures of large RNA molecules. `RNA-MoIP` is publicly available at: `http://csb.cs.mcgill.ca/RNAMoIP`

1.5 Outline

The following chapters are organized as follows. In Chapter 2 we formally define motifs, describe our motif database, introduce our IP constraints and our software `RNA-MoIP`.

In Chapter 3, we apply `RNA-MoIP` on a set of 9 RNA used by [11] to benchmark RNA 3D structure prediction programs. Our results show that `RNA-MoIP` (i) improves the accuracy of canonical base pair predictions , (ii) identifies the best secondary structures in a pool of suboptimal structures generated by `RNAsubopt`, and (iii) predicts accurate 3D structures for sequences with sizes varying between 53 and 128 nucleotides – an insight that cannot be reached by other programs.

In Chapter 4, we discuss our results and propose future research directions.

We expend on the softwares that were used for this work: `RNAsubopt`, `Rna3Dmotif` and `MC-Sym`, in Appendix A.

`RNA-MoIP` is publicly available at: `http://csb.cs.mcgill.ca/RNAMoIP`.

CHAPTER 2
Methods

Let ω be an RNA sequence. First, we use a classical secondary structure predictor (e.g. `RNAsubopt`) to generate a list of sub-optimal secondary structures. Second, for each structure from the list we use `RNA-MoIP` to insert RNA 3D motifs in the structure using the sequence information provided by ω. `RNA-MoIP` works in two steps:

1. Given a database of sequences of RNA 3D motifs (cf. Section 2.2), the preprocessing step applies a classical pattern matching algorithm to find all occurrences of each motif in the input sequence ω.
2. Given this list of potential insertion sites and a secondary structure, we solve an integer programming (IP) problem which minimizes our objective function (cf. Section 2.3). Importantly, under certain conditions, `RNA-MoIP` allows base pair removals to insert the 3D motifs.

Finally, we use the best solutions as templates for `MC-Sym` [17] and generate three-dimensional structures. In particular, we force `MC-Sym` to use the motifs inserted by `RNA-MoIP` at their predicted location, instead of letting `MC-Sym` build his own solution for the motifs. As we will see later, these constraints enable us to produce 3D structures, when an unconstrained run would simply never end.

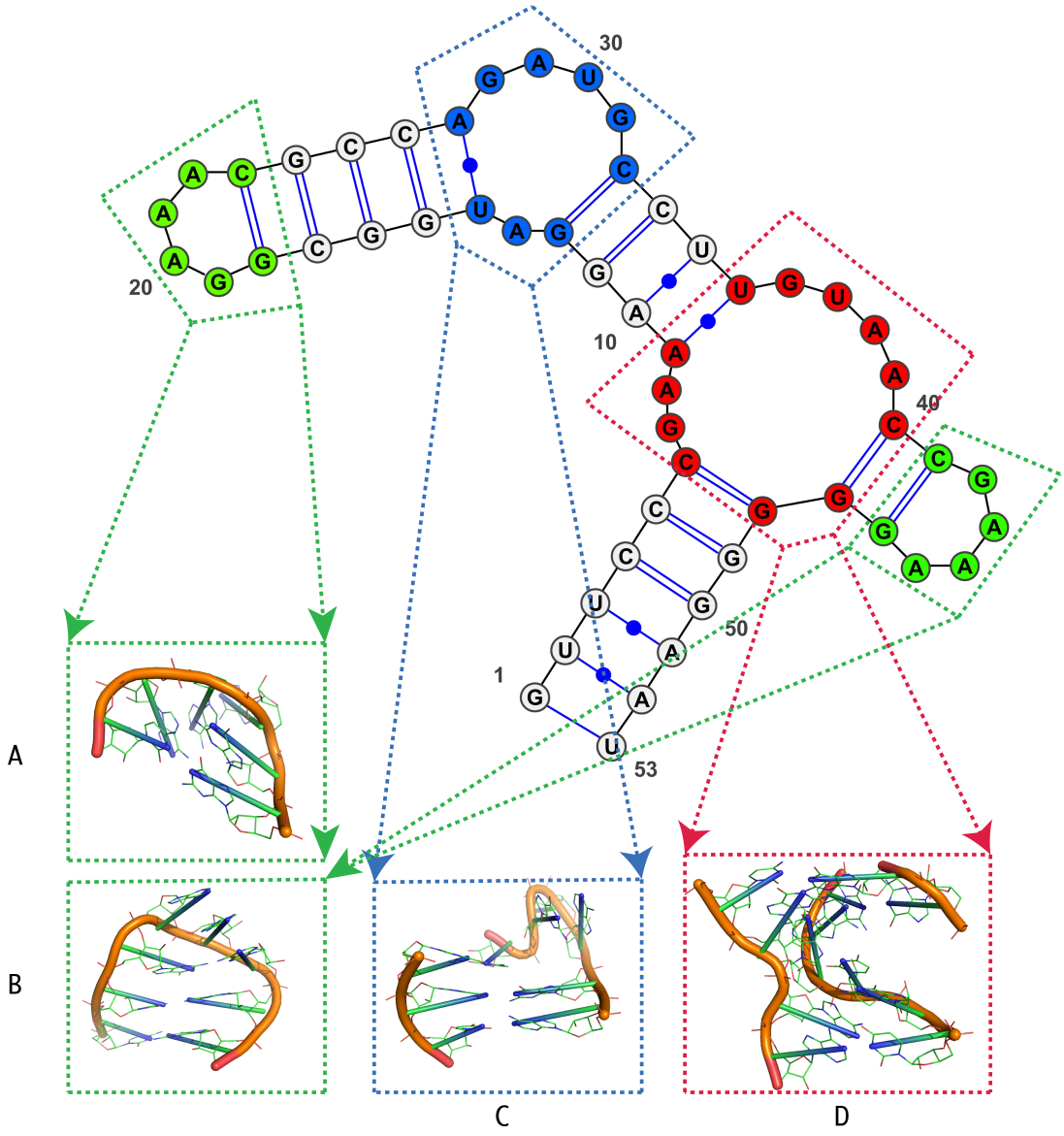


Figure 2–1: Example of motifs extraction

This is an example of motifs extracted by `Rna3Dmotif` [4] from a given RNA. When our framework receives these as input it defines the following. The hairpins form the group with one component and we write: $A := [(\text{GGAAAC})]$, $B := [(\text{CGAAAG})]]$. Interior loops and Bulges, have two components (e.g. $C := [(\text{GAU}), (\text{AGAUGC})]$). The n-way junctions naturally have n components. In this case there is a 3-way junction which can be written in our framework in two ways: $D := [(\text{CGAA}), (\text{UGUAAC}), (\text{GG*})]$ or $D := [(\text{CGAA}), (\text{UGUAAC}), (\text{*GG})]$, since we want components to be of size at least 3. D can also be written as $M^D := \text{CGAA} - \text{UGUAAC} - \text{GG*}$ (resp. $\text{CGAA} - \text{UGUAAC} - \text{*GG}$) and we can say that D match this sequence at $(6, 35, 47)$ but also, $(6, 35, 48)$ and many other positions. A motif can be inserted multiple times.

2.1 Definitions

Motif: We represent a *motif* x as an ordered list of components (i.e. sequences) where x_i^j is the i-th nucleotide of the j-th component (i.e sequence). As presented in Fig. 2–1, hairpins have one component, bulges and internal loops have two, and k-way junctions have k. Let r be the number of components, we represent a motif as $x := [(x_1^1, \cdots, x_{k_1}^1), \cdots, (x_1^r, \cdots, x_{k_r}^r)]$ and $x_i^j \in \{A, U, G, C, *\}$ where $*$ represents a wildcard. We say that motif x if of order r since it has r components. We also write a motif x as:

$$M^x := x_1^1 \cdots x_{k_1}^1 - x_1^2 \cdots x_{k_{r-1}}^{r-1} - x_1^r \cdots x_{k_r}^r$$

i.e. the concatenations of its letters with the added character "$-$" between the components. We define $|M^x|$ as the number of nucleotides in x.

Match: Given a sequence $\omega \in \{A, U, G, C\}^+$, and a motif x with r components, we say that ω_i is the i-th character of ω, and that a motif x *matches* the sequence ω at $(p_1, \cdots, p_r)$ if $\forall\ 1 \leq i < r :\ p_i + k_i + 5 \leq p_{i+1}$ and $\forall\ j \in \{1, \cdots, r\} i \in \{1, \cdots, k_j\}$: $x_i^j \equiv \omega_{p_j+i-1}$ where the p_i's indicate the first positions of the i-th component of motif x in ω. The inequality ensures that each component is separated by at least 5 nucleotides. The number of components inside a motif is crucial to our technique. Since there is not yet a good definition of motifs and components, this allows to consolidate the notion that components interact together but from a certain distance.

2.2 RNA Motifs Database

Here we describe how we build the motifs database. First, we retrieve 888 experimentally determined RNA three-dimensional structures from the Protein Data Bank [1] (`www.pdb.org`). Then, we use the program `RNA3Dmotifs` from [4] to extract all the motifs from these structures. This results in a dataset of 35724 motifs for which we have a 3D *pdb* file and a description of the interactions.

We processed these data to create a non-redundant database of curated motifs. In order to ensure the compactness and coherency of the motifs, we assume that each component is at least 5 nucleotides farther than the previous one; otherwise the nucleotides are merged in a single component and the missing positions are replaced by a wildcard "$*$" (See Sec. 2.1). We describe a motif m returned by `RNA3Dmotifs` as $m := \{(m_1, p_1), \cdots, (m_n, p_n)\}$, where $m_k \in \{\mathrm{A}, \mathrm{U}, \mathrm{G}, \mathrm{C}\}$, $p_k < p_{k_+1} \in \mathbb{N}$ and p_k is the position of nucleotide m_k in the sequence it was extracted from. We create x such that if we set $x_i^j = m_k$ and $1 \leq p_{k+1} - p_k = \alpha < 5$ then $\forall\ i < i' < i + \alpha :\ x_{i'}^j = *$ and $x_{i+\alpha}^j = m_{k+1}$. If $p_{k+1} - p_k \geq 5$ then $x_1^{j+1} = m_{k+1}$.

Some motifs may have small components composed of one or two nucleotides. In our framework, the insertion of these components will be less constrained by the secondary structure and thus less specific. To avoid this case, we extend all small components in all possible combinations with the character $*$ until they reach a size of three (e.g. the last component of motif D in Fig. 2–1).

It is worth noting that all these constraints are empirical rules which aim to remove discrepancies and unify the sequence constraints applied on motifs. They

should not be considered as a rigid framework but rather as a tentative to clarify the `RNA3Dmotifs` output.

Finally, we cluster together all pairs of motifs x, y if $M^x \equiv M^y$ (i.e. the sequences are identical) to obtain a non sequence-redundant database of 4708 motifs.

It is important to note that in our database, the motifs with one single component are all hairpins and do not include bulges. In this work, bulges will be seen as a particular case of interior loops since for the motif to loop, it needs to include the complementary strand.

Finally, we excluded from this database the structures used in the benchmark (See Sec.3.2).

2.3 Integer programming model

Here, we describe the integer programming equations used to insert the motifs into a given secondary structure. To insert a motif into the structure, our model allows some base pairs to be removed.

2.3.1 Input

We introduce the notation and sets that will be used to model our input data. Let ω be a RNA sequence, and S a secondary structure of ω without pseudo-knots. We denote by $n = |\omega|$ the length of ω, and by δ the maximum percentage of base pairs that is allowed to be removed. We call B the set of base pairs found in the secondary structure S. We denote by Mot^j the set of motifs with j components that match ω:

$$Mot^j = \{x \mid x := [(x_1^1, \cdots, x_{k_1}^1), \cdots, (x_1^j, \cdots, x_{k_j}^j)] \text{ and } \exists \text{ a } match \text{ of } x \text{ in } \omega\} \quad (2.1)$$

We store in Seq_i^j the positions where the i-th component of the motifs of order j can be inserted:

$$\begin{aligned} Seq_i^j = \{(x, p_i, p_i + k_i - 1) \mid x \in Mot^j \text{ and} \\ \exists \text{ a } match \ (p_1, \cdots, p_{i-1}, p_i, p_{i+1}, \cdots, p_j) \text{ of } x \text{ in } \omega\} \end{aligned} \quad (2.2)$$

We note that the criteria used to determine whether a motif can be inserted is based on the sequence only. At this stage, the secondary structure S is not used.

2.3.2 Variables

We now describe the two variables used in our model. Our program will make two predictions: First, it finds the location of the insertion sites of the motifs, and second, it predicts which base pairs are removed. We denote $C_{k,l}^{x,j}$ as the boolean variable indicating the insertion of the j-th component of the motif x between positions k and l in ω. Similarly, we use the boolean variable $D_{u,v}$ to indicate if the base pair $(u, v) \in B$ is removed or not (i.e. $D_{u,v} = 1$ if (u, v) is removed from the secondary structure S and 0 otherwise).

2.3.3 Objective function

We describe here the optimization criteria that will be used to predict the insertion of the RNA motifs. As mentioned earlier, we do not have any estimate of the energy of the motifs retrieved with `RNA3Dmotifs`. Instead, we will use a principle of minimum entropy. We assume that a molecule folds in a configuration that stabilizes its backbone and side chains through various base pairings. In other words, we aim to minimize the free variables of the molecule. In the absence of reliable energy values, we assign to the motifs a weight equivalent to the square of the number of nucleotides in its components. This objective function aims to increase the coherency of the motif insertions as it maximizes the nucleotide positions coverage and favours the insertion of large motifs instead multiple small ones. It also eases the $3D$ reconstruction process with `MC-Sym`. Although this objective function is purely heuristic, it performed well in this work. We give a penalty of 10 for every base pair deleted. With lower values our model was removing as many base pairs as possibles, while with higher values we obtained similar results. Formally, we aim to minimize the following function:

$$10 * \sum_{(u,v)\in B} D_{u,v} - \sum_{x\in Mot^j} \left((|M^x|)^2 \cdot \sum_{(x,k,l)\in Seq_1^j} C_{k,l}^{x,1} \right) \tag{2.3}$$

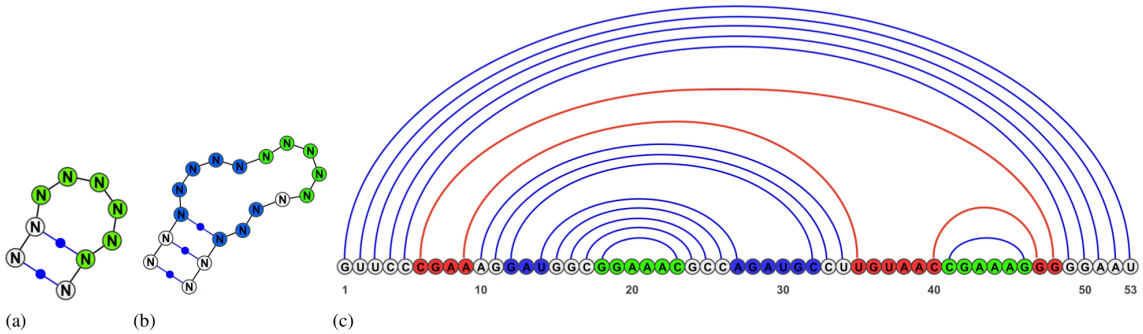


Figure 2–2: Constraints configurations

2.3.4 Constraints

Here, we describe the constraints that we use to ensure the correctness of the motif insertion and to control the coherency of the final structure. We detail these equations below.

Hairpins.

$$\forall (x,k,l) \in Seq_1^1 : \; C_{k,l}^{x,1} \leq \sum_{\substack{(u,v)\in B \\ k-1\leq u\leq k \wedge l\leq v\leq l+1}} (1 - D_{u,v}) \; + \sum_{\substack{(\tilde{x},\tilde{k},\tilde{l})\in Seq_1^2 \\ \tilde{l}=k-1}} C_{\tilde{k},\tilde{l}}^{\tilde{x},1} \; + \sum_{\substack{(\tilde{x},\tilde{k},\tilde{l})\in Seq_2^2 \\ \tilde{k}=l+1}} C_{\tilde{k},\tilde{l}}^{\tilde{x},2} \quad (2.4)$$

We use Constraint (2.4) to insert the hairpins (i.e. $x \in Mot^1$). A hairpin can be inserted if and only if one of two following criteria holds: A base pair $(u,v) \in B$ exists such that both extremities are stacked or overlap on the motif x (Fig.2–2a), or there is an inserted motif y with two components (i.e. $y \in Mot^2$) such that x is nested inside y and stacked onto one of its components (Fig. 2–2b).

Interior Loops & Bulges.

$$\forall (u,v) \in B,\ \forall x \in Mot^2 :\ -n \cdot D_{u,v} \leq \sum_{\substack{(x,k,l) \in Seq_1^2 \\ l<u \vee v<k}} C_{k,l}^{x,1} - \sum_{\substack{(x,k,l) \in Seq_2^2 \\ l<u \vee v<k}} C_{k,l}^{x,2} \leq n \cdot D_{u,v} \quad (2.5)$$

$$\forall (x,k,l) \in Seq_1^2, \forall (x,\tilde{k},\tilde{l}) \text{ s.t. } \left[\tilde{k} > l \wedge 2 \sum_{\substack{(u,v) \in B \\ k \leq u \leq l \wedge \tilde{k} \leq v \leq \tilde{l}}} 1 \ + \sum_{\substack{(u,v) \in B \\ k \leq u \leq l \oplus \tilde{k} \leq v \leq \tilde{l}}} 1 \geq l - k + \tilde{l} - \tilde{k} + 1 \right] \in Seq_2^2 :$$

$$C_{k,l}^{x,1} + C_{\tilde{k},\tilde{l}}^{x,2} \leq 1 \quad (2.6)$$

Constraints (2.5) and (2.6) are used to insert bulges and interior loops. Constraint (2.5) stipulates that for all base pairs $(u,v) \in B$, every motif in Mot^2 must have as many 1-st component inserted before u or after v, as it has 2-nd components, allowing to create an arc between the components of every motif without creating a pseudo-knot with the base pairs in the secondary structure. Constraint (2.6) allows both components to be inserted only if they fill at least 2 unpaired positions. Indeed, such insertion would most likely not produce valuable structural information.

k-way junctions.

$$\sum_{\substack{j \geq 3 \\ (x,k,l) \in Seq_1^j}} C_{k,l}^{x,1} \leq 1 \quad (2.7)$$

$$\forall j \geq 3,\, \forall (u,v) \in B:\; -n \cdot D_{u,v} \leq (j-1) \cdot \sum_{\substack{(x,k,l) \in Seq_1^j \\ u \leq k \leq l \leq v}} C_{k,l}^{x,1} \; - \sum_{\substack{1 < i \leq j \\ (x,k,l) \in Seq_i^j \\ u \leq k \leq l \leq v}} C_{k,l}^{x,i} \leq n \cdot D_{u,v} \quad (2.8)$$

Constraints (2.7) and (2.8) describe how k-way junctions are inserted. Constraint (2.7) restricts the number of inserted motifs with three or more components to one, which is a reasonable assumption given the size of the RNAs. Combined with (2.8), it means that for every conserved base pair $(u,v) \in B$, a motif can be inserted if all or none of the components are between u and v. This is equivalent, as we can see in Fig. 2–2c, to saying that we can connect the components which are shown in red without creating a pseudo-knot with the base pairs in the secondary structure.

Motifs completeness.

$$\forall\, 1 \leq i < j,\; \forall (x,k,l) \in Seq_i^j:\; C_{k,l}^{x,i} \leq \sum_{\substack{(x,\tilde{k},\tilde{l}) \in Seq_{i+1}^j \\ l+5 < \tilde{k}}} C_{\tilde{k},\tilde{l}}^{x,i+1} \quad (2.9)$$

$$\forall\, 1 < i \leq j,\; \forall (x,k,l) \in Seq_i^j:\; C_{k,l}^{x,i} \leq \sum_{\substack{(x,\tilde{k},\tilde{l}) \in Seq_{i-1}^j \\ \tilde{l} < k-5}} C_{\tilde{k},\tilde{l}}^{x,i-1} \quad (2.10)$$

$$\forall\, j > 1,\; \forall x \in Mot^j,\; \forall 1 < i \leq j: \sum_{(x,k,l) \in Seq_1^j} C_{k,l}^{x,1} - \sum_{(x,\tilde{k},\tilde{l}) \in Seq_i^j} C_{\tilde{k},\tilde{l}}^{x,i} = 0 \quad (2.11)$$

Constraints (2.9) , (2.10) and (2.11) ensure that the insertions of the components in ω respect their order given in the motif. Constraints (2.9) and (2.10) require that if $C_{k,l}^{x,j}$ is the j-th component of motif x and it is inserted at positions k, l, then at least one $(j-1)$-th component of the same motif should be inserted 5 nucleotides above, and one $(j+1)$-th component after. The last constraint restricts that, since a motif can be inserted many times, the multiplicity of every component should be equal to the multiplicity of the 1-st component.

Secondary Structure Constraints.

$$\forall j > 1, \forall 1 \leq i \leq j, \forall (x,k,l) \in Seq_i^j : \; C_{k,l}^{x,i} \leq \sum_{\substack{(u,v)\in B \\ k-1\leq u\leq k\vee \\ l\leq u\leq l+1\vee \\ k-1\leq v\leq k\vee \\ l\leq v\leq l+1}} (1 - D_{u,v}) \tag{2.12}$$

$$\forall 1 \leq u \leq n : \sum_{\substack{(x,k,l)\in Seq_i^j \\ k<u<l}} C_{k,l}^{x,i} + \frac{1}{4} \sum_{\substack{(k,l)\in B \\ k=u\vee l=u}} (1 - D_{k,l}) \; + \frac{3}{4} \sum_{\substack{(x,k,l)\in Seq_i^j \\ k=u\vee l=u}} C_{k,l}^{x,i} \leq 1 \tag{2.13}$$

$$\forall 1 < u < n : (1 - \sum_{\substack{(\tilde{u},\tilde{v})\in B \\ \tilde{u}=u-1\vee\tilde{v}=u-1}} D_{\tilde{u},\tilde{v}}) - (1 \; - \sum_{\substack{(\tilde{u},\tilde{v})\in B \\ \tilde{u}=u\vee\tilde{v}=u}} D_{\tilde{u},\tilde{v}}) + (1 - \sum_{\substack{(\tilde{u},\tilde{v})\in B \\ \tilde{u}=u+1\vee\tilde{v}=u+1}} D_{\tilde{u},\tilde{v}}) \geq 0 \tag{2.14}$$

$$\sum_{(i,j)\in B} D_{i,j} \leq \delta \cdot |B| \tag{2.15}$$

We conclude by describing the constraints regulating the secondary structure properties. Constraint (2.12) use the secondary structure to guide the sites of the components by allowing insertions if and only if one extremity overlaps or is stacked on top of a base pair. Constraint (2.13) forbids two components from overlapping to each other, and prevents base pairs to occur inside a component. Constraint (2.14) uses the formulation of [18] to prevent lonely base pairs (i.e. every position in a base pair must also have an adjacent position in a base pair). Constraint (2.15) limits the number of canonical base pairs δ of S that can be removed.

We can notice that the model always has a trivial feasible solution, when no motifs are inserted and no base pairs removed.

CHAPTER 3
Results

3.1 Implementation

To solve the IP problem, we use the Gurobi optimizer v.4.5.1 [9] API for Python. We ran our benchmark on a Ubuntu-Server 10.04 on a Dell PE T610 2x Intel Quad core X5570 Xeon Processor, 2.93GHz 8M Cache, 64GB Memory (8x8GB), 1333MHz Dual Ranked RDIMMs for 15 Processors, Advanced ECC.

3.2 Data set

We validate our method on the dataset defined by [11] to benchmark the RNA 3D structure prediction programs.. In this work, we aim to predict the structure of large RNAs with 3-way and 4-way junctions. Small sequences (less than 50 nucleotides) with simpler structures can be accurately predicted using existing methods such as `MC-Pipeline` or `NAST`. Therefore, we removed from the dataset sequences with less than 50 nucleotides. We also removed RNAs with secondary structures that include pseudo-knots. Indeed, our approach has been designed to use secondary structures predicted by classical secondary structure predictors such as `RNAfold` and `RNAstructure`, thus without pseudo-knots. Moreover, our motif database and IP model have not been designed to insert pseudo-knots. We redirect the reader interested in application of IP techniques to the prediction of pseudo-knots to the recent

works of [18] and [23]. Our final dataset includes eleven RNAs with sequences of lengths ranging from 53 to 128 nucleotides. We note that two of these 11 had no homologous 3-way junctions in our database. We present here the results on the remaining 9 RNAs. Eight of them have a 3-way junction and the other a 4-way junction. Importantly, the motifs extracted from these RNAs by `RNA3Dmotifs` [4] have been removed from our motif database.

For the negative control test, we used a test set composed of the 24 RNAs from the dataset defined by C. Laing and T. Schlick [11] without pseudo-knot, 3 or 4-way junction. Their sizes range from 16 to 77 nucleotides.

3.3 `RNA-MoIP` pipe-line

Our RNA tertiary structure prediction pipe-line works in three steps. First, secondary structures are predicted using classical predictors such as `RNAfold`, `RNAstructure` or `unafold`. In this work, we generated the input secondary structures with `RNAsubopt` [27]. We used the default parameters but discarded structures with lonely pairs (i.e stems of length 1). This procedure generated between 1 and 22 secondary structures for each RNA sequence. Nonetheless, the quality of secondary structure predictions is too low on the *riboswitch* $3D2G$ from *A. thaliana* and the *tRNA* $2DU3$ from *A. fulgidus* to accommodate k-way junction motifs insertion. Therefore, we extended our list of suboptimal structures and generated all secondary structures in the range of 4.5 kcal/mol from the *mfe*. This operation resulted in a total of 242 (resp. 58) secondary structures. We also note that extending the list of suboptimal structures of other RNAs produces identical results. Typically the secondary structure predictors

generate lists of suboptimal structures from which it is difficult to extract the best ones. We will see in Sec. 3.3.2 that our method is able to identify the best candidates in these ensemble predictions.

We apply `RNA-MoIP` to insert RNA 3D motifs in these secondary structures as described in Sec. 2.3. The solution with an optimal score, under our objective function (Sec. 2.3.3) is scripted manually for `MC-Sym` with the motifs locked in. Due to various `MC-Sym` features, it is currently difficult to generate automatically these scripts. Hence, the processing of very large sequence data sets remains challenging. We recall that many $3D$ structures can have the same motif, which is only determined by the sequence. Here we provide all alternative configurations to `MC-Sym`. Time is a major limitation of `MC-Sym`. Using our strategy, we show that preprocessing the sequences with `RNA-MoIP` results in a dramatic time improvement and at the same time improves the accuracy. We set a time limit of 30min. Then on every set of predicted structures we apply a minimization of *steepest-descent* until the difference of energy between two consecutive structures is smaller than 5 Kcal/mol/A or after 500 steps [17]. It is worth noting that `MC-Sym` was not able to generate a structure in two cases (3E5C and 2GDI), although `RNA-MoIP` predicted the 3-way junctions at the correct positions.

3.3.1 Negative control

We verify that `RNA-MoIP` does not predict wrong k-way junction insertions (i.e. false positives). We use a negative control data set composed of the 24 RNAs extracted from the dataset of C. Laing and T. Schlick that contain hairpins and interior

loops motifs but without pseudo-knots and k-way junctions. Then, we apply the protocol described as in Sec. 3.3. Our results indicate that no k-way junction have been inserted in the optimal solution returned by `RNA-MoIP`.

3.3.2 Secondary Structure

The identification of the best secondary structures in a list of suboptimals is one major challenge in RNA secondary structure ensemble prediction. We present in Table 3–1 the results as follows. The first column shows the PDB identifier of the RNAs. The two following columns show the ratio of well predicted base pairs in two structures. The former is the structure in the optimal solution of the RNA and the latter is the same structure after `RNA-MoIP` was applied and removed some base pairs (we highlight in bold the improved scores). There is an average increase of 6% and only one case where it decreased. The fourth column represents the average of well predicted base pairs for each RNA over all secondary structures considered by `RNA-MoIP`. The penultimate column shows the rank of the best secondary structure selected by `RNA-MoIP` in the ordered list of suboptimal secondary structures generated with `RNAsubopt`, while the last column shows the total number of suboptimal secondary structures in that list. As we can see, the average base pair accuracy of the secondary structure prediction is approximately 63%. But when we look at the base pair accuracy of the secondary structures selected by `RNA-MoIP` (78%) we observe a major improvement of 15% which means that our approach is able to identify the best secondary structures in a pool of candidates. Interestingly, our program is able to extract candidates with a very low rank, when ordered by energy. For instance, on 2DU3 `RNA-MoIP` extracts the 163-th candidate with a base pair accuracy of 91%

	Percentage of well predicted base pairs in the predicted secondary structures			Secondary structure selected by `RNA-MoIP`	
PDB	Optimal solution		Average over all	Rank in	Nb. of candidate
	Before	After	secondary structures	`RNAsubopt` list	secondary structures
3E5C	100	100	100	1	2
1DK1	88	**92**	82	1	7
1MMS	47	**67**	49	2	2
2DU3	79	**100**	44	52	58
3D2G	91	**100**	43	163	243
2HOJ	68	68	61	13	20
2GDI	96	94	71	10	22
1LNG	100	100	82	1	7
1MFQ	29	**31**	31	1	4
Average	78	**84**	63		

Table 3–1: Secondary structure improvement

(vs. 43% in average) in a pool of 258 structures. Finally, to accommodate motif insertions `RNA-MoIP` can remove base pairs. Once removed, the ratio of well predicted base pairs reaches 84%, thus increases by 6%. This experiment demonstrates that the insertion of motifs can help to identify incorrectly predicted base pairs.

3.3.3 Three-dimensional Structure

We evaluate the quality of our 3D structure predictions using the RMSD and the *RNA Interaction Network Fidelity* [7] tool, available with the `MC-Pipeline` at

PDB	NTs	RNA-MoIP		Nb. 3D	MCC			RMSD			Structure
		Sec. structs.	time (s)		Max	Avg	SD	Min	Avg	SD	
3E5C	53	2	0.27	0	–	–	–	–	–	–	3-way (riboswitch)
1DK1	57	7	3.11	106	0.88	0.81	0.03	2.95	4.76	0.99	3-way
1MMS∗	58	2	0.31	105	0.68	0.63	0.03	5.66	7.65	0.86	3-way
2DU3∗	71	58	139.96	7	0.90	0.82	0.04	2.23	2.91	0.44	4-way (tRNA)
3D2G∗#	77	243	1268.63	8	0.84	0.80	0.02	5.34	7.35	1.34	3-way (riboswitch)
2HOJ∗#	79	20	27.44	155	0.84	0.80	0.01	3.19	7.29	2.31	3-way (riboswitch)
2GDI∗	80	22	47.1	0	–	–	–	–	–	–	3-way (riboswitch)
1LNG∗	97	7	110.96	146	0.88	0.84	0.02	2.73	6.30	1.91	3-way (SRP)
1MFQ∗	128	4	46.06	14	0.77	0.72	0.03	9.07	14.34	5.01	3-way (SRP)

Table 3–2: Three-dimensional structure prediction evaluation

major.iric.ca/MC-Pipeline/. The latter computes the true positive (TP), false positive (FP) and false negative (FN) tertiary structure interactions between the experimental structures deposited in the PDB [1] and our predictions, and returns the positive predictive valuablee (PPV) and sensitivity (STY) defined as:

$$PPV := \frac{|TP|}{|TP| + |FP|} \qquad STY := \frac{|TP|}{|TP| + |FN|}.$$

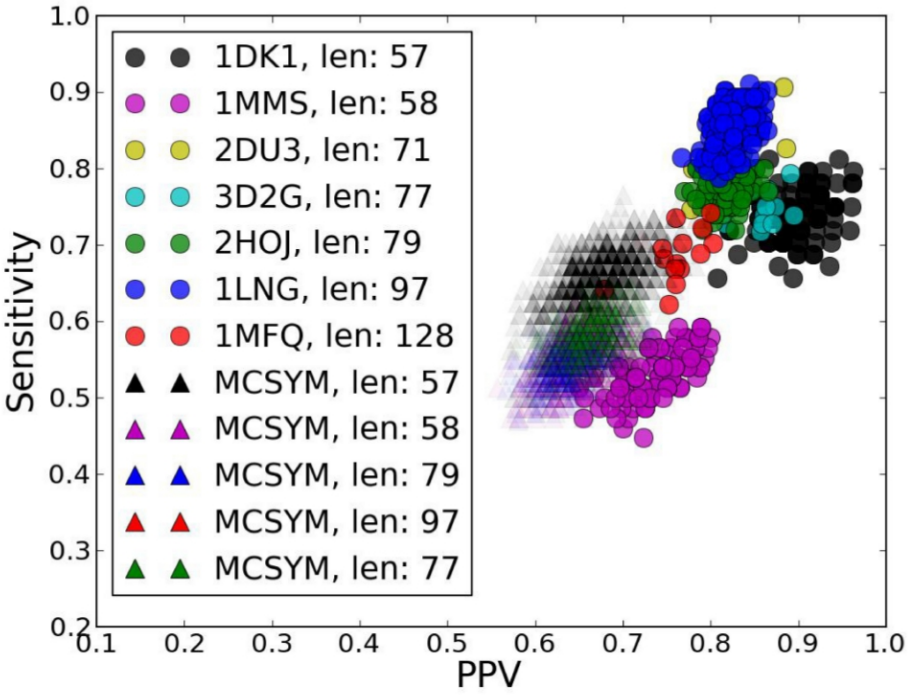
We report our results in Fig. 3–1. Fig. 3–1b shows that RNA-MoIP coupled with MC-Sym is able to predict most of the tertiary structure interactions.

We show in green in Fig. 3–1b the RMSDs of the solutions obtained with MC-Sym as described in 3.3. We recall that each script for MC-Sym is done manually with the positions inside the predicted motifs directly mapped to the pool of corresponding $3D$ structures, obtained by RNA3Dmotifs from [4]. We also recall that [11] reported that only the two smallest structures were resolved by MC-Fold | MC-Sym pipeline when only the sequence was given. We thus decided to input into MC-Sym the sequence with the secondary structure selected by RNA-MoIP. Under this scenario, MC-Sym was allowed to run for 48 hours. Those results are shown in blue. As we can see, having the secondary structures allowed to solve 5 of the 7 structures. We note that two of them only produced 7 solutions in the first half hour. The largest one took more then 4 hours to produce the first results, and had only two solutions after the 48 hours. Nonetheless the information given by the motifs allows to our method to predict significantly more accurate results.

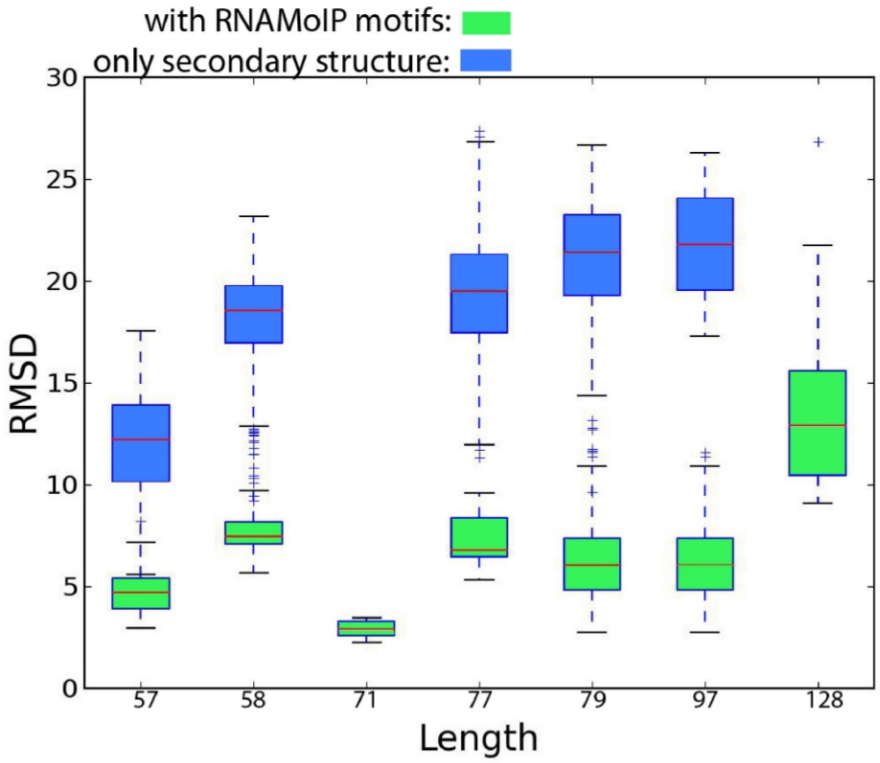
Fig. 3–2 shows that our program outperforms other software and produces 3D structures with a RMSD significantly lower than those observed by [11] for other

programs. It also shows that our method scales with the length of the RNA better than other approaches.

We completed this analysis by computing the Matthews Correlation Coefficient (MCC), defined as: $MCC := \sqrt{\text{PPV} \times \text{STY}}$, and the running time of our method. We show in Table 3–2 an overview of the results obtained on each RNA as follows. In the first column RNA identifiers are followed by a "$*$" or a "#" to denote that `MC-Sym` (reps. `NAST`) failed to predict them, as reported by [11]. The second column contains the length of each RNA. The third column contains the number of secondary structure predicted by `RNAfold` and used as input for `RNA-MoIP`. The fourth column is the total time (preprocessing and solve) in seconds taken by `RNA-MoIP` to find an optimal solution for all the secondary structures. The fifth column contains the number of $3D$ structures generated by `MC-Sym` with the script made with `RNA-MoIP` optimal solution. We then have the maximal, average, and standard deviation of the MCC. The following three columns present the minimal, average, and standard deviation of the RMSD. Finally, the last column indicate the type of junction found in the native structure. We note the fast execution time of `RNA-MoIP`, even when a large number of secondary structures are used. Also, despite a time limit of 30min, `MC-Sym` generates good candidates. Noticeably, our RMSD can be as low as 2.23Å for the tRNA 2DU3 and are considerably smaller than those reported by [11] (See Fig. 3–2).



(a)



(b)

Figure 3–1: PPV, STY and RMSD of predicted three-dimensional structures

Fig. 3–1a shows the PPV and STY for all the 3D structures generated by our scripts on `MC-Sym` against the reference on the PDB [1]. Fig. 3–1b shows in green the distribution of the RMSD of the solutions obtained with `MC-Sym` when the structures of the motifs were given and in blue when only the secondary structure was provided. N.B.: In the latter, the molecules are identified by their size.

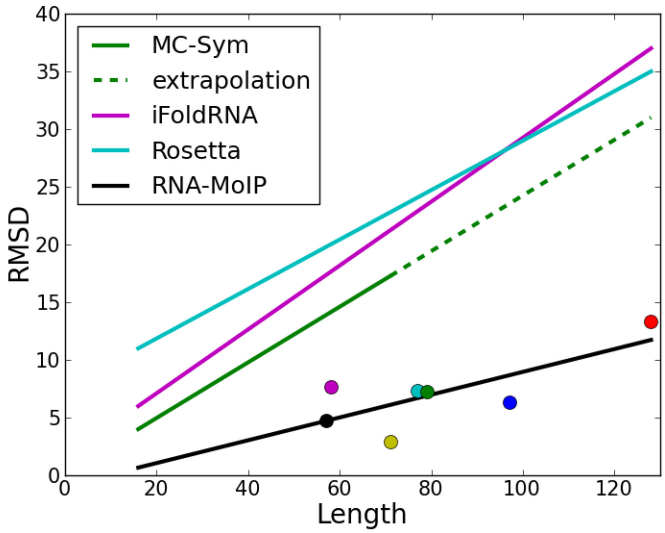


Figure 3–2: Predictors performance comparison

Comparison of the RMSD obtained by `RNA-MoIP`, `MC-Pipeline`, `iFoldRNA` and `FARNA` by [11]. This figure is derived from the data computed by [11] on which we superposed the results obtained by `RNA-MoIP` and `MC-Sym`. The dots are the average RMSD shown in Fig. 3–1b. We also show in dotted line the extrapolated RMSD for `MC-Sym` and in black the best fit for the average RMSD obtained with our pipeline.

CHAPTER 4
Conclusion

In this thesis, we demonstrated that large RNA 3D structures can be automatically predicted using a hierarchical approach. We benefit of the progresses accumulated over the last 30 years in the field of RNA secondary structure prediction. We developed an IP framework to incorporate the novel local motifs information available in databases. We showed that this approach enables us to predict very quickly accurate 3D structures for large RNA sequences (more than 50 nucleotides). By contrast, previous methods were either too slow or too inaccurate on molecules with similar sizes.

We show that motif insertion enables us to identify the best secondary structures in a pool of suboptimal structures. Nonetheless, the choice of the size of the sample set that needs to be generated remains an open problem. As illustrated by the $3D2G$ and $2DU3$ experiments, some RNAs may require significantly more suboptimal structures than those generated by default by `RNAsubopt`. A simple strategy to reduce the search space would be to cluster those samples and pick representative structures.

`RNA-MoIP` demonstrates that we can already benefit from the information accumulated in RNA local motif databases without deriving a new model. This is

important because the paucity of the data currently available in these databases prevents from developing accurate statistical potentials for predicting tertiary structure interactions in high-order motifs such as the k-way junctions.

Therefore, another important issue with `RNA-MoIP` is the completeness of the motif database. For instance, we have seen that there is no homologous 3-way junction in our database that can be correctly inserted in 3EGZ (riboswitch in *H. sapiens*) and 2OIU (synthetic ribozyme). To circumvent this limitation, an interesting approach would be to generate highly probable new motifs from the existing ones using isostericity matrices [25].

Some of the IP techniques developed here could be implemented using a dynamic programming paradigm. However, we argue that the IP approach is more flexible and more suited to this problem. Indeed, in our framework the rules of insertions can be easily modified (i.e. adding, removing or changing an equation) while a dynamic programming scheme would required a complete re-implementation. This is particularly useful in this case where some motifs present in our databases have specific insertion constraints. This situation is more likely to happen in the future with the growth of RNA local motif databases. Moreover, we demonstrated in this work that our implementation is fast enough for realistic applications.

Finally, our methods are compatible with state-of-the-art IP programs for pseudoknot predictions [23, 18]. In future work, we could envision to merge the two models and include new rules for inserting highly sophisticated 3D motifs with long-range interactions, coaxial stacking or base triplets. Beside its inherent flexibility, the

development of IP models for RNA structure prediction finds a justification in recent results showing the inapproximability of the prediction of RNA pseudo-knotted secondary structures with a nearest neighbour model [24].

Appendix A
Software

A.1 `RNAsubopt`

Our method relies on our ability to predict quickly a set of secondary structure yielding with high probability a good candidate, as discussed in Chapter 2. A well accepted and efficient tool to this end is `RNAsubopt` [26], part of the Vienna RNA Package [8]. `RNAsubopt` uses many thermodynamic parameters to compute the minimal free energy (*mfe*) secondary structure and than retrieve all secondary structures within a given energy range from the *mfe*.

The thermodynamic parameters defines the energy for the basic pieces of the secondary structures (e.g. base pairs, stacking, bulges, loops, etc...) and can be finely tuned by hand. However, our method, as explained in Chapter. 2, relies on having quickly a pool of secondary structures. `RNA-MoIP` then identifies the one deemed interesting. The focus was thus in having the secondary structure of interest inside the pool, the ranking (i.e. the energy of the structure) is not taken into consideration by `RNA-MoIP`. As such, when default parameters did not produce a valid secondary structure, we expanded the range allowed range of energy from the *mfe* to 4.5 kcal/mol (Chapter 3).

A.2 Rna3Dmotif

A few databases of RNA 3D motifs have been done, as briefly discussed in Sec. 1.3. At first, methods relied on the nucleotides spatial position (e.g. [6]) to identify structural motifs in RNA. We chose the tool `Rna3Dmotif` [4] which, rather than spatial positions, considers the following three types of interactions:

1. The phosphodiester bonds linking nucleotides along the backbone.
2. The canonical base pairs forming the secondary structure (i.e. Watson-Crick GC,AU and wobble GU).
3. The 12 non-canonical base pairs defined by Leontis and Westhof [14].

`Rna3Dmotif` relies on manual annotation of the interaction of RNAs three-dimensional structures by `FR3D` [22]. Those structur1es are in the Protein Data Bank [1] (`www.pdb.org`). `Rna3Dmotif` uses the `FR3D` annotation to identify structural motifs and the `.pdb` file to extract every structural motif. For every motif two files will be created. The former contains a description of every interaction in the motif. The latter contains the spacial positions of all atoms in the motif in a canonical `.pdb` file.

Our framework uses the two files as discussed in Chapter 2. In the first step, `RNA-MoIP` uses the description of the interaction to identify the sequences of the components and where the motifs can be inserted. In the second step, the `.pdb` files are feed to `MC-Sym` to do the 3D predictions.

A.3 MC-Sym

MC-Sym (Macromolecular Conformations by SYMbolic programming) [17] is the most accurate tool to predict all-atoms RNA 3D structures as shown in [11]. This tool is based around the notion of NCMs (nucleotide cyclic motifs) which are minimal structural motifs. It is important to notice that as much as 100 different 3D structures can be associate to any NCM. The basic idea of MC-Sym is to assemble those NCMs under all possible configurations that doesn't violate a set of constraints. Those constraints can be explicitly given, as the secondary structure which contains information about nucleotides interactions. There is also a set of implicit constraints, as a minimal distance between atoms and limits in the backbone torsion.

Formally, an MC-Sym script needs a set sequence, an explicit declaration of every needed NCM and the order under which those pieces must be merged. We show in the Figure. A–1 the MC-Sym script that would correspond to trying to model the hairpin shown in Figure. A–2.

For every nucleotide in every NCM, we need to explicitly specify in the sequence which is the corresponding position. We can note in the script how this become tedious when there is no corresponding NCM, as for the hairpin (lines 14 to 28 in Fig. A–1, nucleotides 6 to 10 in Fig. A–2). Those parts are also the hardest to model due to the sparsity of constraints. Nonetheless, this is perfect for the inclusions of the motifs as described in Appendix. A.2. If we had the simple hairpin motif M := CAAACAG and its associated 3D structure as a pdb file, we could have directly included it in the MC-Sym script as:

```
motif_M = library(
```

```
pdb( "Path/To/Motifs/M/*.pdb" )
#1:#7 <- A5:A11)
```

Therefore, once we have a database of 3D motifs and we known the insertion locations of motifs into a given RNA, it is quite straightforward to create the `MC-Sym` script associated with it. Notice that a motif is always defined as a set of files (i.e. `*.pdb`). Nonetheless, the order in which the different NCMs and motifs are merged is crucial to the diversity and feasibility of the simulation. There seems to be no fixed reason for why some combination are more efficient than others, so scripting must be done by trial and error. In our work, we always limited the time to half and hour, and limited the number of generated structures to 1000, since speed and accuracy are crucial to a useful tool.

```
01  sequence( r A1  ACUGCAAACAGCACG )
02  ncm_1 = library(
03      pdb( "Path/To/Motifs/ACCG/*.pdb.gz" )
04      #1:#2, #3:#4 <- A1:A2, A14:A15)
05  ncm_2 = library(
06      pdb( "Path/To/Motifs/CUAC/*.pdb.gz" )
07      #1:#2, #3:#4 <- A2:A3, A13:A14)
08  ncm_3 = library(
09      pdb( "Path/To/Motifs/UGCA/*.pdb.gz" )
10      #1:#2, #3:#4 <- A3:A4, A12:A13)
11  ncm_4 = library(
12      pdb( "Path/To/Motifs/GCGC/*.pdb.gz" )
13      #1:#2, #3:#4 <- A4:A5, A11:A12)
14  ncm_5 = library(
15      pdb( "Path/To/Motifs/CA/*.pdb.gz" )
16      #1:#2, #3:#4 <- A5:A6)
17  ncm_6 = library(
18      pdb( "Path/To/Motifs/AA/*.pdb.gz" )
19      #1:#2, #3:#4 <- A6:A7)
20  ncm_7 = library(
21      pdb( "Path/To/Motifs/AA/*.pdb.gz" )
22      #1:#2, #3:#4 <- A7:A8)
23  ncm_8 = library(
24      pdb( "Path/To/Motifs/AC/*.pdb.gz" )
25      #1:#2, #3:#4 <- A8:A9)
26  ncm_9 = library(
27      pdb( "Path/To/Motifs/CA/*.pdb.gz" )
28      #1:#2, #3:#4 <- A9:A10)
29  structure = backtrack
30  (
31   ncm_1
32   merge( ncm_2 1.5 )
33   merge( ncm_3 1.5 )
34   merge( ncm_4 1.5 )
35   merge( ncm_5 1.5 )
36   merge( ncm_6 1.5 )
37   merge( ncm_7 1.5 )
38   merge( ncm_8 1.5 )
39   merge( ncm_9 1.5 )
40  )
41
42  explore
43  (
44      structure
45      option(
46      model_limit = 1000,
47      time_limit = 30m,
48      seed =  42 )
49  )
```

Figure A–1: `MC-Sym` script modelling a simple hairpin

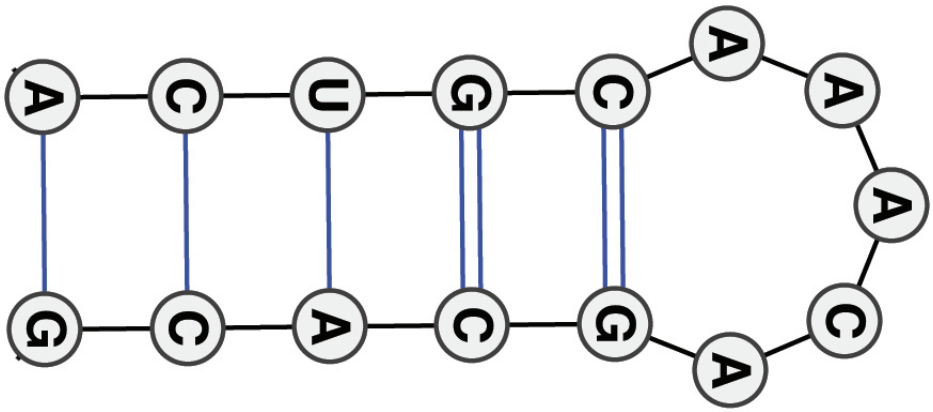


Figure A–2: Simple hairpin

A simple hairpin composed of 4 stacked base pair NCMs and 5 free nucleotides

References

[1] H M Berman, J Westbrook, Z Feng, G Gilliland, T N Bhat, H Weissig, I N Shindyalov, and P E Bourne. The protein data bank. *Nucleic Acids Res*, 28(1):235–42, Jan 2000.

[2] Eckart Bindewald, Robert Hayes, Yaroslava G Yingling, Wojciech Kasprzak, and Bruce A Shapiro. RNAJunction: a database of RNA junctions and kissing loops for three-dimensional structural analysis and nanodesign. *Nucleic Acids Res*, 36(Database issue):D392–7, Jan 2008.

[3] José Almeida Cruz and Eric Westhof. Sequence-based identification of 3D structural modules in rna with rmdetect. *Nat Methods*, 8(6):513–21, Jun 2011.

[4] Mahassine Djelloul and Alain Denise. Automated motif extraction and classification in RNA tertiary structures. *RNA*, 14(12):2489–97, Dec 2008.

[5] Chuong B Do, Daniel A Woods, and Serafim Batzoglou. CONTRAfold: RNA secondary structure prediction without physics-based models. *Bioinformatics*, 22(14):e90–8, Jul 2006.

[6] Carlos M. Duarte. RNA structure comparison, motif search and discovery using a reduced representation of RNA conformational space. *Nucleic Acids Research*, 31(16):4755–4761, August 2003.

[7] Patrick Gendron, Sébastien Lemieux, and François Major. Quantitative analysis of nucleic acid three-dimensional structures. *Journal of molecular biology*, 308(5):919–36, May 2001.

[8] Ivo L Hofacker. RNA secondary structure analysis using the Vienna RNA package. *Curr Protoc Bioinformatics*, Chapter 12:Unit12.2, Jun 2009.

[9] Inc Houston, Texas: Gurobi Optimization. Gurobi optimizer version 4.5.1. Software Program, 2011.

[10] Fabrice Jossinet, Thomas E Ludwig, and Eric Westhof. Assemble: an interactive graphical tool to analyze and build RNA architectures at the 2D and 3D levels. *Bioinformatics*, 26(16):2057–9, Aug 2010.

[11] Christian Laing and Tamar Schlick. Computational approaches to 3D modeling of RNA. *J Phys Condens Matter*, 22(28):283101, Jul 2010.

[12] Alexis Lamiable, Dominique Barth, Alain Denise, Franck Quessette, Sandrine Vial Vial, and Eric Westhof. *Automated prediction of three-way junction topological families in RNA secondary structures*. Computational Biology and Chemistry, 2012.

[13] Sébastien Lemieux and François Major. RNA canonical and non-canonical base pairing types: a recognition method and complete repertoire. *Nucleic Acids Res*, 30(19):4250–63, Oct 2002.

[14] Neocles B. Leontis and Eric Westhof. Geometric nomenclature and classification of RNA base pairs. *Rna*, 7(04):499–512, 2001.

[15] Nicholas R Markham and Michael Zuker. UNAFold: software for nucleic acid folding and hybridization. *Methods Mol Biol*, 453:3–31, 2008.

[16] Hugo M Martinez, Jacob V Maizel, Jr, and Bruce A Shapiro. RNA2D3D: a program for generating, viewing, and comparing 3-dimensional models of RNA. *J Biomol Struct Dyn*, 25(6):669–83, Jun 2008.

[17] Marc Parisien and François Major. The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data. *Nature*, 452(7183):51–5, Mar 2008.

[18] Unyanee Poolsap, Yuki Kato, and Tatsuya Akutsu. Prediction of RNA secondary structure with pseudoknots using integer programming. *BMC Bioinformatics*, 10 Suppl 1:S38, 2009.

[19] Vladimir Reinharz, Fran¸cois. Major, and Jérôme Waldispühl. Towards 3D structure prediction of large RNA molecules: an integer programming framework to insert local 3D motifs in RNA secondary structure. *Bioinformatics*, 28(12):i207–i214, June 2012.

[20] Jessica S Reuter and David H Mathews. RNAstructure: software for RNA secondary structure prediction and analysis. *BMC Bioinformatics*, 11:129, 2010.

[21] Winston Salser. Globin mRNA Sequences: Analysis of Base Pairing and Evolutionary Implications. *Cold Spring Harbor Symposia on Quantitative Biology*, 42:985–1002, January 1978.

[22] Michael Sarver, Craig L Zirbel, Jesse Stombaugh, Ali Mokdad, and Neocles B Leontis. FR3D: finding local and composite recurrent structural motifs in RNA 3D structures. *J Math Biol*, 56(1-2):215–52, Jan 2008.

[23] Kengo Sato, Yuki Kato, Michiaki Hamada, Tatsuya Akutsu, and Kiyoshi Asai. IPknot: fast and accurate prediction of RNA secondary structures with pseudoknots using integer programming. *Bioinformatics*, 27(13):i85–93, Jul 2011.

[24] Saad Sheik, Rolf Backofen, and Yann Ponty. Impact of the energy model on the complexity of RNA folding with pseudoknots. In *Proceedings of the 23rd Annual Symposium on Combinatorial Pattern Matching*, 2012.

[25] Jesse Stombaugh, Craig L Zirbel, Eric Westhof, and Neocles B Leontis. Frequency and isostericity of RNA base pairs. *Nucleic Acids Res*, 37(7):2294–312, Apr 2009.

[26] Stefan Wuchty, Walter Fontana, Ivo L Hofacker, and Peter Schuster. Complete Suboptimal Folding of RNA and the Stability of. *Biopolymers*, 49:145–165, 1999.

[27] Stephan Wuchty, Walter Fontana, Ivo L Hofacker, and Peter Schuster. Complete suboptimal folding of RNA and the stability of secondary structures. *Biopolymers*, 49(145-165), 1999.

[28] Shay Zakov, Yoav Goldberg, Michael Elhadad, and Michal Ziv-Ukelson. Rich parameterization improves rna structure prediction. In *RECOMB*, pages 546–562, 2011.

[29] Christian Höner zu Siederdissen, Stephan H Bernhart, Peter F Stadler, and Ivo L Hofacker. A folding algorithm for extended RNA secondary structures. *Bioinformatics*, 27(13):i129–36, Jul 2011.

[30] Michael Zuker and Patrick Stiegler. Optimal computer folding of lare RNA sequences using thermodynamics and auxiliary information. *Nucleic acids research*, 9(1):133–148, 1981.