Evaluation of a Sophisticated SCFG Design for RNA Secondary Structure Prediction

Markus E. Nebel, Anika Scheid[*,†]
Department of Computer Science, University of Kaiserslautern, Germany
{nebel,a_scheid}@cs.uni-kl.de

Abstract

Motivation: Predicting secondary structures of RNA molecules is one of the fundamental problems of and thus a challenging task in computational structural biology. Over the past decades, mainly two different approaches have been considered to compute predictions of RNA secondary structures from a single sequence: the first one relies on physics-based and the other on probabilistic RNA models. Particularly, the free energy minimization (MFE) approach is usually considered the most popular and successful method. Moreover, based on the paradigm-shifting work by McCaskill which proposes the computation of partition functions (PFs) and base pair probabilities based on thermodynamics, several extended partition function algorithms, statistical sampling methods and clustering techniques have been invented over the last years. However, the accuracy of the corresponding algorithms is limited by the quality of underlying physics-based models, which include a vast number of thermodynamic parameters and are still incomplete. The competing probabilistic approach is based on stochastic context-free grammars (SCFGs) or corresponding generalizations, like conditional log-linear models (CLLMs). These methods abstract from free energies and instead try to learn about the structural behavior of the molecules by learning (a manageable number of) probabilistic parameters from trusted RNA structure databases.
In this work, we introduce and evaluate a sophisticated SCFG design that mirrors state-of-the-art physics-based RNA structure prediction procedures by distinguishing between all features of RNA that imply different energy rules. This SCFG actually serves as the foundation for a statistical sampling algorithm for RNA secondary structures of a single sequence that represents a probabilistic counterpart to the sampling extension of the PF approach. Furthermore, some new ways to derive meaningful structure predictions from generated sample sets are presented. They are used to compare the predictive accuracy of our model to that of other probabilistic and energy-based prediction methods.
Results: Particularly, comparisons to lightweight SCFGs and corresponding CLLMs for RNA structure prediction indicate that more complex SCFG designs might yield higher accuracy but eventually require more comprehensive and pure training sets. Investigations on both the accuracies of predicted foldings and the overall quality of generated sample sets (especially on an abstraction level, called *abstract shapes* of generated structures, that is relevant for biologists) yield the conclusion that the Boltzmann distribution of the PF sampling approach is more centered than the ensemble distribution induced by the sophisticated SCFG model, which implies a greater structural diversity within generated samples. In general, neither of the two distinct ensemble distributions is more adequate than the other and the corresponding results obtained by statistical sampling can be expected to bare fundamental differences, such that the method to be preferred for a particular input sequence strongly depends on the considered RNA type.

1 Introduction

The function of an RNA molecule in the cell's metabolism is often to a large extend determined by its structure. Since the experimental determination of the complete 3D structure of a molecule, called its *tertiary structure*, is usually time-consuming and expensive, and its prediction is computationally complex, it has proven convenient to first search for its 2D structure, called the *secondary structure* of the molecule. In fact, most of the 3D conformation is given by the intramolecular base pairings in the plane and thus, it is customary for prediction algorithms to allow only non-crossing (nested) base pairs given by the secondary structure, such that the molecule can be modeled as a planar graph [Wat78].

[*] Corresponding author.

[†] The research of the author is supported by the Carl-Zeiss-Stiftung.

In structural biology, the most successful and still most appreciated techniques for the computational prediction of RNA secondary structure from a single sequence are based on thermodynamic models and use the free energy minimization (MFE) paradigm to identify candidate structures for the given RNA sequence. All these algorithms are realized by dynamic programming (DP) routines that run in $\mathcal{O}(n^3)$ time and require $\mathcal{O}(n^2)$ storage for a sequence of length n. While early methods, like [NPGK78, NJ80, ZS81], computed only one structure (the MFE structure of the molecule), several efficient algorithms have been developed over the years for generating a set of suboptimal foldings (see e.g. [WFHS99, Zuk89]). Widely used implementations of such MFE based algorithms are for instance the Mfold software [Zuk89, Zuk03] or the Vienna RNA package [HFS$^+$94, Hof03]. However, the quality of such physics-based methods is strongly dependent on and thus limited by the used thermodynamic models.

For the standard sequence-dependent thermodynamic model for RNA secondary structures, usually called *Turner model*, free energy parameters and rules have been estimated for basic structural motifs [XSB$^+$98, MSZT99], but there are still substantial uncertainties in the corresponding comprehensive free energy parameters. Actually, since the considered thermodynamic parameters are mostly estimated from experimental results the rules for computing the energies of particular substructures are still incomplete. In particular, extrapolations are currently used for large loops.

Moreover, it is practically impossible to incorporate information on folding kinetics, as certain important chemical aspects (like for example the influence of proteins/enzymes or the effect of co-transcriptional folding) can simply not be measured in terms of free energy. As a consequence, although the Turner model is considered valid for any type of RNA, it encounters specific problems for particular types of RNA (e.g., for tRNAs where it is well-known that modified nucleotides introduce problems for structure prediction [RCM99]).

One way to overcome these problems is to estimate the thermodynamic parameters from RNA structure databases via Bayesian statistical inference (where the experimentally derived Turner parameter values can be used for prior specification) [Din06]. In fact, such a Bayesian inference approach not only makes it possible to derive energy estimates that are suited for structure prediction. If applied to a training set of RNA data from a single biological class it may also manage to indirectly incorporate non-energetic effects (like, e.g., modified nucleotides) into the model, since those are observed in the trusted training set and thus may alter the energy parameters derived. In any case, the accuracy of the estimated parameters strongly depends on the quality of the employed data.

Another way to overcome difficulties in connection with MFE structures is the partition function (PF) approach for computing base pair probabilities as introduced in [McC90], providing a statistical characterization of the equilibrium ensemble of RNA secondary structures. On its basis, a statistical sampling algorithm as implemented in the Sfold software [DL03, DCL04]) can be used to generate a structurally diverse set of suboptimal foldings which – compared to the set of structurally quite similar suboptimal structures usually computed by MFE based DP algorithms – can be much closer to the structure determined by comparative analysis [Din06]. Note that Sfold actually predicts suboptimal foldings as *centroids* of clusters of candidate structures obtained from statistical sampling (by employing precomputed base pairing probabilities) rather than from an MFE based DP traceback. However, if only the optimal (MFE) structure is needed, a strict DP variant should be preferred in terms of the running time. Nevertheless, since the PF – and thus sampling based on it – is dependent on free energies, it is however also limited by the underlying thermodynamic model. In fact, as the most probable structures in the Boltzmann-weighted ensemble are equal to the MFE (or something close to it) structures, this approach inherits some of the problems associated with traditional MFE approaches. As one consequence, Bayesian inference of energy parameters is also used in connection with PF based sampling approaches.

An alternative methodology towards single sequence RNA secondary structure prediction is based on modeling the class of all feasible secondary structures (that obey to certain structural constraints like for example the non-existence of isolated base pairs) by *stochastic context-free grammars (SCFGs)*, which induce a (non-uniform) probability distribution on the considered class. Particularly, being an extension of usual context-free grammars (CFGs), SCFGs do not only model the class of objects (language) to be generated, but also define a joint probability distribution on them. In a sense, this SCFG approach can be seen as a generalization of hidden Markov models, which are widely and successfully used in the large field of bioinformatics. In fact, when using SCFG based approaches, the main focus of attention is laid on the typical structural composition of foldings and free energies are disregarded. An example for a popular SCFG based prediction tool for RNA secondary structure is Pfold [KH99, KH03].

As there is no *lab-based* prior to the grammar parameters like the Turner model for MFE and PF approaches, the corresponding distribution has to be derived from a collection of real-life RNA data (RNA sequences with known secondary structures) when using probabilistic[1] approaches to RNA structure pre-

[1] In this paper we call an approach *probabilistic* if it makes no use of free energy based models; even if a PF based

diction. This for example can be done by counting the observed frequencies of applications of the distinct production rules of an unambiguous SCFG (yielding a maximum likelihood estimator), by expectation maximization or similar methods from machine-learning. That way, the resulting estimates of the grammar parameters are adapted to the considered data set. Again, we have two different choices: First, we may consider a training set where only structures of a single biological class (e.g., tRNA) are contained. Here, we may expect that all structural properties (including aspects which are caused by interaction with proteins or by other *non-energetic* details of RNA folding) typical to this class are trained into the respective parameter values. For a general model of RNA folding, this may lead to lack of generalization since we cannot be sure that the model adapts well to new data from a different class. Second, we may use a rich training set of mixed biological classes. Here, the before mentioned danger is much smaller but we lose the chance to capture some class-specific properties of the structures within our model. In both cases, the main problem that comes inherently with the SCFG approach for modeling RNA structures and limits the performance of the corresponding computational prediction methods is that it is obviously highly dependent on the availability of a rich, reliable training set. This is especially the case when using a complex SCFG design that distinguishes between all different features in RNA structure aiming at a highly realistic model for there a large number of parameters needs to be determined.

Early probabilistic approaches such as [KH99] seem to have chosen the structure of their SCFG rather arbitrarily; at least, there is almost no discussion about the motivation for the choice of the productions. This problem has first been addressed in [DE04] where nine different SCFGs have been evaluated in connection with RNA secondary structure prediction. Aiming at an exploration on how different SCFG designs affect the accuracy of single sequence RNA secondary structure prediction methods, the authors observed that fairly simple SCFGs achieve respectable prediction accuracies, but – despite the uncertainties in Turner's energy model – the best physics-based methods still generally perform significantly better than the best SCFGs. Therefore, the authors of [DE04] raised the following questions, which will be addressed by this article:

1) Could an appropriately designed sophisticated SCFG be able to outperform the existing MFE methods for single sequence prediction?

2) How would an (unambiguous)[2] SCFG mirroring state-of-the-art physics-based algorithms (i.e. a grammar with specific productions for all structural motifs for which there are different thermodynamic parameters or energy rules) perform?

As already noted, in order to improve the predictive accuracy of energy-based algorithms, (some of) the corresponding thermodynamic parameters might be estimated or improved via statistical inference methods, by taking advantage of a particular RNA database. This obviously strongly relates to the estimation of the grammar parameters of a sophisticated SCFG design as described in question 2). Actually, if a certain energy parameter value for a specific structural motif can be statistically estimated from a given set of real-world RNA data, then the corresponding grammar parameter for the production that generates this motif can effectively be trained from the same data set, yielding a one-to-one correspondence between estimated thermodynamic and grammar parameter values. Hence, it might be assumed that a sophisticated SCFG satisfying the conditions formulated in question 2) has a similar predictive power than modern physics-based algorithms that employ elaborate free energy models.

According to these aspects, it should also be mentioned that recently, a new RNA secondary structure prediction tool named CONTRAfold [DWB06] has been introduced, which is based on a flexible probabilistic model, called *conditional log-linear model (CLLM).* CLLMs are a generalization of traditional SCFGs according to the following facts: While SCFGs (like hidden Markov models) are *generative* probabilistic models, which are intuitive and allow convenient *generative* parameter training via maximum *joint likelihood* techniques, CLLMs are *discriminative* probabilistic models, where the parameters are learned by *discriminative* training which maximizes the *conditional likelihood.* As stated in [DWB06], any SCFG has an equivalent representation as an appropriately parameterized CLLM. The prime advantage of using CLLMs instead of vanilla SCFGs (i.e. discriminate instead of generative training) is that CLLMs have the power to represent more complex scoring schemes than the corresponding SCFG can represent. In fact, CONTRAfold uses a simplified Mfold-like scoring scheme for the underlying CLLM providing a rather high single sequence prediction accuracy and closing the performance gap between the best physics-based and the best probabilistic RNA structure prediction methods.

Boltzmann sample is a random event, we accordingly do not assume it probabilistic.

[2] A structurally ambiguous SCFG mirror of modern energy-based algorithms for single sequence structure prediction has already been described in [RE00].

Moreover, due to the previously mentioned benefit caused by departing from the common MFE approach to considering the sampling extension of the PF approach, it seems reasonable to rely on Boltzmann samples rather than on single MFE structures in order to address question 1). Accordingly, we decided to oppose the Boltzmann samples to corresponding samples obtained by a SCFG version of Sfold's statistical sampling strategy based on an appropriately designed grammar that actually meets the requirements raised in question 2). This means we will employ an efficient statistical sampling algorithm that incorporates comprehensive structural features and – instead of the recent thermodynamic Turner parameters – additional information obtained from trusted databases of real-world RNA structures in order to generate probabilistic counterparts of the Boltzmann samples. Actually, just like in the PF variant, secondary structures are sampled rigorously from the ensemble distribution of all feasible foldings for a given input sequence, but the distribution will be induced by the parameter values of the underlying SCFG.
Altogether, due to the before mentioned connection of thermodynamic parameters and probabilities of a sophisticated grammar (especially if both are estimated statistically), it seems adequate to put the following hypothesis which will be examined within this article:

H_0: The Boltzmann distribution implied by a thermodynamic PF approach and the ensemble distribution induced by a corresponding (sophisticated) SCFG are similar and thus yield comparable statistical sampling results (i.e. no significant differences of the generated sample sets can be expected).

According to the preceding explanations, the main objectives of this paper are given as follows: We will answer the two important questions 1) and 2) already raised in [DE04] (according to the previously mentioned aspects) and essentially check whether hypothesis H_0 can be verified. Therefore, we will first define a sophisticated SCFG that represents a probabilistic mirror to the optimization schemes applied in modern MFE based dynamic programming routines and statistical sampling approaches based on free energies and PFs. Actually, that SCFG is designed to represent an exact probabilistic mirror to the diverse recursions and formulae for calculating all equilibrium PFs and sampling probabilities that are needed for the elaborate statistical sampling procedure applied in the Sfold software.
Another take on the same kind of problems but with slightly different intensions can be found in [RLE11]. There, in order to explore a range of probabilistic models of increasing complexity, and to directly compare probabilistic, thermodynamic, and discriminative approaches, a computational tool is created that can parse a wide spectrum of RNA grammar architectures (including the standard nearest-neighbor model and more) using a generalized super-grammar that can be parameterized with probabilities, energies, or arbitrary scores. The authors put forward that discriminative training is not required, simple ML learning is enough. Therefore, their tool uses only generative training, not discriminative. Parameters can, however, be imported from other sources. Using their tool Rivas et al. show that probabilistic nearest-neighbor models perform comparably to (but not significantly better than) discriminative methods and that complex statistical models are prone to overfitting RNA structure.
The rest of the present paper is organized as follows: Section 2 describes the SCFG model for secondary structures that will be used as the foundation for the probabilistic sampling approach. The complete sampling strategy is introduced in Section 3 and Section 4 proposes several appropriate ways for deriving particular predictions from generated structure samples. Notably, some of them deal with a new mechanism for controlling the prediction accuracy (by a sensitivity/PPV trade-off parameter $\gamma_{\mathrm{t-o}}$) similar to the one implemented in the CONTRAfold software. Section 5 examines the benefits and potential drawbacks of using a sophisticated SCFG like ours compared to lightweight SCFGs and corresponding CLLMs for RNA structure prediction. We find that using a more complex SCFG design might actually yield a higher prediction accuracy but requires a more comprehensive and pure training set to ensure that all parameters are appropriately estimated. To address hypothesis H_0, Section 5 additionally discusses the potentials and pitfalls of the SCFG based sampling method compared to the sampling extension of the PF approach as implemented in the Sfold software, where both the quality of generated sample sets and their applicability to the problem of RNA structure prediction are investigated. These comparisons include results on an abstraction level (*abstract shapes* of sampled structures, as introduced in [JRG08]) that is of great interest and relevance for biologists. One of the prime observations is that the SCFG induced distribution implies a greater structural diversity within generated samples, as it seems to be less centered than the Boltzmann energy distribution. Moreover, the distinct comparisons indicate that using a lean database of mixed RNA classes results in improper estimators of the needed grammar parameters, such that in these cases the PF approach usually generates more realistic samples. The SCFG approach generally produces more accurate sample sets if a rich and pure training set is available. In summary, free energy based samplers are proven to have stronger abilities for generalization or vize-versa, approaches based on a sophisticated SCFG can be fitted to a specific class of RNA (where they show high predictive

accuracy possibly implied by *non-energetic* effects which find their way into the parameter set) without generalization to other biological classes (maybe because there those effects behave differently) and thus may be assumed overfitted. However, in Paragraph 5.2.3 we disprove this assumption in the context of tRNA data by showing that our sophisticated SCFG approach does not tend to predict significantly more often a cloverleaf structure than the PF variant. Finally, Section 6 summarizes our findings and hints at some interesting matters for further research.

2 Used SCFG Model

While RNA sequences are usually modeled as strings over the alphabet $\{a, c, g, u\}$, for secondary structures, lots of different representations and corresponding definitions are used in literature. Here, we decided to rely on the following definition:

Definition 2.1 ([ZMT99])**.** A *secondary structure* of size n is a finite set (possibly empty) of *base pairs*. A base pair between i and j, $1 \leq i < j \leq n$, is denoted by $i.j$ (or $r_i.r_j$ to stress that the secondary structure is for sequence r). A few constraints are imposed:

1. Two base pairs, $i.j$ and $i'.j'$ are either identical, or else $i \neq i'$ and $j \neq j'$.
2. Pseudoknots (given by two base pairs $i.j$ and $i'.j'$ such that $i < i' < j < j'$) are prohibited.
3. Hairpin loops of size less than $\min_{HL} \geq 1$ are prohibited, i.e. $(j - i - 1) \geq \min_{HL}$ for any pair $i.j$.

RNA secondary structures according to Definition 2.1 can be modeled as strings over the alphabet $\{(,), \circ\}$, where a dot $\circ$ represents an unpaired nucleotide and a pair of corresponding brackets **( )** represents two bases in the RNA molecule that are paired (see [VC85]). Using this dot-bracket representation, we can easily model sequences and secondary structures[3] as formal languages $\mathcal{L}_r$ and $\mathcal{L}_s$, respectively, defined by corresponding (stochastic) context-free grammars that generate them. Usual CFGs are only capable of modeling the elements of a formal language, whereas SCFGs can be used to additionally define a probability distribution on its words (or their derivation trees). A formal definition is given as follows:

Definition 2.2 ([FH72])**.** A *stochastic context-free grammar (SCFG)* is a 5-tuple $G = (I, T, R, S, \mathrm{Pr})$, where I (resp. T) is an alphabet (finite set) of intermediate (resp. terminal) symbols (I and T are disjoint), $S \in I$ is a distinguished intermediate symbol called *axiom*, $R \subset I \times (I \cup T)^*$ is a finite set of production rules and Pr is a mapping from R to $[0, 1]$ such that each rule $f \in R$ is equipped with a probability $p_f := \mathrm{Pr}(f)$. The probabilities are chosen in such a way that for all $A \in I$ the equality $\sum_{f \in R} p_f \cdot \delta_{Q(f),A} = 1$ holds. Here, δ is Kronecker's delta and $Q(f)$ denotes the source of the production f, i.e. the first component A of a production rule $(A, \alpha) \in R$. In the sequel, we will write $p_f : A \to \alpha$ instead of $f = (A, \alpha) \in R$, $p_f = \mathrm{Pr}(f)$.

We assume the reader to be familiar with the basic definitions and concepts regarding SCFGs. For a fundamental introduction on stochastic context-free languages, see for example [HF71]. Nevertheless, it is worth mentioning that if a formal language is modeled by a so-called *consistent* SCFG, then the probability distribution on the production rules of the SCFG implies a probability distribution on the words of the generated language and thus on the modeled structures[4].

For the prediction of RNA secondary structures, SCFGs are used in the following way: a suitable (ambiguous) grammar models the combinatorial class (language) of all RNA sequences (i.e., this SCFG generates all possible primary structures), while each derivation tree for a given sequence uniquely corresponds to one possible secondary structure. Therefore, a grammar at least has to distinguish between paired and unpaired positions by using different productions to generate the corresponding symbols of the RNA sequence. However, it is possible to use a grammar which generates paired and unpaired positions located in different kinds of substructures (like hairpin loops or bulges) by different production rules. That way one aims at a more realistic model since it becomes possible to use different probabilities within the different contexts (substructures).

According to our objectives motivated in Section 1, our SCFG should be constructed to represent a mirror to the free energy model employed in Sfold's sampling procedure, which means we have to take care of the

[3] Note that in order to avoid ambiguity, we will denote a particular sequence by r and the corresponding secondary structure by s in the sequel.

[4] To ensure that a SCFG gets consistent, one can for example assign relative frequencies to the productions, which are computed by counting the production rules used in the leftmost derivations of a finite sample (RNA database) of words from the generated language [CPG83].

fact that all distinct structural features of RNA for which there are different energy rules and free energy parameters according to the underlying thermodynamic model have to be modeled by corresponding distinct production rules. Briefly, at any point, the desired SCFG must be capable of distinguishing between exactly the same mutually exclusive and exhaustive cases that have to be considered in the recursions for calculating the equilibrium PFs as defined in [DL03]. Then, the inside and outside values derived for a given sequence on the basis of that SCFG can be used in a straightforward fashion – along with the corresponding SCFG parameters (rule probabilities) – in order to define the needed conditional sampling probabilities that directly correspond to those applied in Sfold's elaborate PF based sampling algorithm. By using different intermediate symbols for the distinct loop types and their respective substructures, we obtain the following sophisticated SCFG design for modeling the formal language $\mathcal{L}_s$ of all RNA secondary structures:

Definition 2.3. The (unambiguous) SCFG $\mathcal{G}_s$ generating exactly the language $\mathcal{L}_s$ is given by $\mathcal{G}_s = (\mathcal{I}_{\mathcal{G}_s}, \Sigma_{\mathcal{G}_s}, \mathcal{R}_{\mathcal{G}_s}, S)$, where $\mathcal{I}_{\mathcal{G}_s} = \{S, T, C, A, P, L, F, H, G, B, M, O, N, U, Z\}$, $\Sigma_{\mathcal{G}_s} = \{(,), \circ\}$ and for $m_h := \min_{HL} \geq 1$ and $m_s := \min_{\text{hel}} \geq 1$, $\mathcal{R}_{\mathcal{G}_s}$ contains exactly the following rules:

$p_1 : S \to T,$ ⇝ initiate exterior loop

$p_2 : T \to C,$ $p_3 : T \to A,$ $p_4 : T \to CA,$ $p_5 : T \to AT,$ $p_6 : T \to CAT,$ ⇝ composition of exterior loop

$p_7 : C \to ZC,$ $p_8 : C \to Z,$ ⇝ strands in exterior loop

$p_9 : A \to (^{m_s} L)^{m_s},$ ⇝ initiate helix

$p_{10} : P \to (L),$ ⇝ extend helix

$p_{11} : L \to F,$ $p_{12} : L \to P,$ $p_{13} : L \to G,$ $p_{14} : L \to M,$ ⇝ initiate any loop

$p_{15} : F \to Z^{m_h - 1} H,$ ⇝ start hairpin loop

$p_{16} : H \to ZH,$ $p_{17} : H \to Z,$ ⇝ extend hairpin loop

$p_{18} : G \to BA,$ $p_{19} : G \to AB,$ $p_{20} : G \to BAB,$ ⇝ type of bulge/interior loop

$p_{21} : B \to ZB,$ $p_{22} : B \to Z,$ ⇝ strands in bulge/interior loop

$p_{23} : M \to UAO,$ ⇝ first substructure of multiple loop

$p_{24} : O \to UAN,$ ⇝ second substructure of multiple loop

$p_{25} : N \to UAN,$ $p_{26} : N \to U,$ ⇝ kth substructure of multiple loop, $k \geq 3$

$p_{27} : U \to ZU,$ $p_{28} : U \to \epsilon,$ ⇝ strands in multiple loop

$p_{29} : Z \to \circ.$ ⇝ unpaired base

The unambiguity of that grammar can be proven along the lines of [NS]. Note that the productions $F \to Z^{m_h - 1} H$ and $A \to (^{m_s} L)^{m_s}$ ensure that neither hairpin loops of less than m_h unpaired nucleotides nor helices of less than m_s consecutive base pairs are generated.

Obviously, the (unambiguous) grammar $\mathcal{G}_s$ can immediately be transformed into a second (ambiguous) SCFG $\mathcal{G}_r$ that models the language $\mathcal{L}_r$ of all RNA sequences: we only have to replace $\Sigma_{\mathcal{G}_s} = \{(,), \circ\}$ by $\Sigma_{\mathcal{G}_r} := \{a, c, g, u\}$ and the three rules $A \to (^{m_s} L)^{m_s}$, $P \to (L)$ and $Z \to \circ$ by corresponding new productions generating *valid*[5] base pairs and unpaired bases, respectively. Finally, in order to guarantee that appropriate probabilities are used for the production rules of the SCFG $\mathcal{G}_r$, we can assign relative frequencies (which can be derived from an arbitrary training set of known RNA sequences with corresponding secondary structures) to the elements in $\mathcal{R}_{\mathcal{G}_r}$, yielding a consistent SCFG.

However, we can equivalently only consider the initial grammar $\mathcal{G}_s$ with *transition probabilities* for the productions in $\mathcal{R}_{\mathcal{G}_s}$ and – in order to be able to model structures on RNA sequences – two additional sets of *emission probabilities* for unpaired bases (i.e., for each $x \in \Sigma_{\mathcal{G}_r}$) and for base pairs (i.e., for every $x_1 x_2 \in \Sigma_{\mathcal{G}_r}^2$). Accordingly, the probability of each production rule in $\mathcal{G}_r$ that generates one or more base pairs ( ) or an unpaired base $\circ$ is given by the product of the corresponding transition probability (for $A \to (^{m_s} L)^{m_s}$, $P \to (L)$ or $Z \to \circ$ in $\mathcal{R}_{\mathcal{G}_s}$) and the respective emission probabilities (for base

[5] Here, we decided to consider any possible pair as valid base pair, where non-canonical ones are mostly prohibited due to small probabilities. Thus, in contrast to the thermodynamics based PF approach which can only handle canonical base pairs, our algorithm is able to deal with arbitrary base pairs, in a convenient way: when using appropriate probabilities, canonical base pairs will be very likely and non-canonical ones will be very unprobable (but not necessarily impossible) to be formed. However, since non-canonical base pairs are usually not permitted in secondary structure models (to limit the number of possible foldings), it would also be adequate to allow only canonical ones. The probabilities for non-canonical base pairs would then be equal to zero.

pairs or unpaired bases)[6]. For example, if $m_s = 2$, then $\Pr(A \to acLgu \in \mathcal{R}_{\mathcal{G}_r}) = \Pr(A \to \boldsymbol{((}L\boldsymbol{))} \in \mathcal{R}_{\mathcal{G}_s}) \cdot \Pr(\text{pair } au) \cdot \Pr(\text{pair } cg)$.
For grammar training this means that instead of using the derivation tree that corresponds to the correct secondary structure s for a given sequence r to determine the relative frequency of each production among all productions with the same premise, we simply have to count the relative frequencies of applications of the production rules of $\mathcal{G}_s$ and the corresponding relative frequencies of emissions of unpaired bases and base pairs that are observed in the training set. The relative frequencies that are obtained in this manner are still a maximum likelihood estimator for the grammar probabilities.
It should be mentioned that the trained transition and emission probabilities are obviously linked in the straightforward mathematical sense, that is the probabilities of the different transitions with same left-hand side, as well as the emissions for unpaired and paired bases, respectively, must sum up to unity. Moreover, all emission probabilities come from the same distribution, that is for any considered loop type, we use the same emission probabilities for unpaired bases located within and base pairs closing a corresponding loop. Consequently, the number of free parameters that have to be trained is given by $\text{card}(\mathcal{R}_{\mathcal{G}_s}) - \text{card}(\mathcal{I}_{\mathcal{G}_s}) + \text{card}(\Sigma_{\mathcal{G}_r})^2 + \text{card}(\Sigma_{\mathcal{G}_r}) = 29 - 15 + 16 + 4 = 34$. Note that this rather moderate number (compared to the heavyweight grammar design) effectively results from linking together the emissions of base pairs generated with different rules instead of going strictly with the grammar definition which implies using different trained distributions for any such rule (here $p_9 : A \to \boldsymbol{(}^{m_s} L \boldsymbol{)}^{m_s}$ and $p_{10} : P \to \boldsymbol{(}L\boldsymbol{)}$). This simplification obviously reduces the dimensionality of the parameter space in a significant way (especially for $\min_{\text{hel}} > 1$), and is also justified due to observations made from considering trusted RNA databases (trained distributions usually are very similar) and having a closer look at the Turner energy parameters (many tables, excluding the stacking table and some others, contain only a few different values in total).

3 Sampling Strategy

In this section, we give a complete derivation of all results needed for a probabilistic statistical sampling algorithm for RNA secondary structures according to the SCFG model defined in the last section. Just like the PF variant, the sampling algorithm has two basic steps: Its forward step computes the inside and outside probabilities for all substrings of an RNA sequence based on the considered SCFG. These inside and outside values are used for calculating *conditional* sampling probabilities for all considered cases. The backward step is basically the same as with PFs, which means it takes the form of a recursive sampling algorithm to randomly draw secondary structures according to the sampling probabilities derived in step one. By applying the algorithm to a biological RNA sequence, a statistically representative sample of secondary structures can quickly be generated once the forward step for deriving the inside and outside values is completed.

3.1 Computing Inside and Outside Probabilities

A detailed description on how the inside and outside variables can be computed with a special variant of an Earley-style parser based on the SCFG $\mathcal{G}_r$ can be found in Section Sm-I[7]. Applying this method to a sequence r of size n, there results cubic time complexity and quadratic memory requirement for the computation of all inside probabilities $\alpha_A(i,j) = \Pr(A \Rightarrow_{lm}^{*} r_i \dots r_j)$ and all outside probabilities $\beta_A(i,j) = \Pr(S \Rightarrow_{lm}^{*} r_1 \dots r_{i-1}\ A\ r_{j+1} \dots r_n)$, $A \in \mathcal{I}_{\mathcal{G}_r}$ and $1 \leq i,j \leq n$.
It should be noted that for the derivation of sampling results presented in Section 5, we actually employed the separation of the grammar parameters into transition and emission probabilities (as explained in Section 2), but for the sake of simplicity, the formal description of the inside and outside algorithms given in Section Sm-I relies on the equivalent unseparated rule probabilities for $\mathcal{G}_r$.

3.2 Sampling Structures According to SCFG Model

Before we will define sampling probabilities for mutually exclusive and exhaustive cases that correspond to those derived in [DL03] with the PF approach, note that when using the PF method, one has to choose a constant value for the parameter $\max_{BL}$ which defines the maximum allowed size of single-stranded regions in bulge and interior loops (for applications, $\max_{BL} = 30$ is a common choice) to

[6] Note that this separation into transition and emission probabilities corresponds to the standard treatment applied in hidden Markov models.

[7] All references starting with Sm are references to the supplementary material available at `http:///wwwagak.cs.uni-kl.de/publications/`.

ensure that the worst-case time complexity remains cubic. However, such restrictions are not necessary to improve the performance of an SCFG based sampling algorithm (see Section Sm-II.1.2), but the corresponding sampling strategy can easily be implemented to deal with $\max_{BL}$, such that (the default value) $\max_{BL} = \infty$ has to be chosen to avoid restrictions on bulge and interior loops.
Nevertheless, we decided to make use of two parameters $\min_{HL}$ and $\min_{\text{hel}}$ to be able to avoid hairpin loops of less than $\min_{HL}$ nucleotides and helices of less than $\min_{\text{hel}}$ consecutive base pairs (such that each paired substructure consists of at least $\min_{\text{ps}} := 2 \cdot \min_{\text{hel}} + \min_{HL}$ bases). This enables us to compare the different results obtained for each combination of the commonly used values $\min_{HL} \in \{1, 3\}$ and $\min_{\text{hel}} \in \{1, 2\}$ to the corresponding results derived with the PF approach (which always implicitly uses $\min_{\text{hel}} = 1$ and $\min_{HL} = 3$).

3.2.1 Sampling Probabilities for Exterior Loops

In the sequel, given an RNA molecule consisting of n nucleotides, we denote the corresponding sequence fragment from position i to position j, $1 \leq i, j \leq n$, by $R_{ij} = r_i r_{i+1} \dots r_{j-1} r_j$.
We start by considering a fragment R_{ij} that does not lie within any regular loop, i.e. that consists only of free bases of the exterior loop. Obviously, we can either leave the whole fragment unfolded or else, we can choose a first free base pair $r_h.r_l$ of the exterior loop (that starts a paired substructure on R_{ij}). As we have to take into account all possible cases for choosing and combining r_h and r_l on the considered fragment, we define $P_0^E(i,j)$ as the sampling probability for leaving R_{ij} single-stranded, $P_{ij}^E(i,j)$ as that for pairing r_i with r_j (i.e., case $h = i$ and $l = j$), $\{P_{hj}^E(i,j,h)\}$ as those for cases where $i < h < l = j$ and $\{P_{il}^E(i,j,l)\}$ as those for cases $h = i < l < j$. Moreover, let $\{P_{hl}^E(i,j,h)\}$ be the probabilities for first sampling h for cases where $i < h < l < j$ and $\{\widehat{P}_{hl}^E(j,h,l)\}$ be those for sampling l after h is sampled (in any case $i < h < l < j$). Using inside outside values and rule probabilities, we find:

$$\begin{aligned}
P_0^E(i,j) &= \frac{1}{p^E(i,j)} \cdot \beta_T(i,j) \cdot (\alpha_C(i,j) \cdot \Pr(T \to C)), \\
P_{ij}^E(i,j) &= \frac{1}{p^E(i,j)} \cdot \beta_T(i,j) \cdot (\alpha_A(i,j) \cdot \Pr(T \to A)), \\
P_{hj}^E(i,j,h) &= \frac{1}{p^E(i,j)} \cdot \beta_T(i,j) \cdot (\alpha_C(i,h-1) \cdot \alpha_A(h,j) \cdot \Pr(T \to CA)), \\
P_{il}^E(i,j,l) &= \frac{1}{p^E(i,j)} \cdot \beta_T(i,j) \cdot (\alpha_A(i,l) \cdot \alpha_T(l+1,j) \cdot \Pr(T \to AT)), \\
P_{hl}^E(i,j,h) &= \frac{1}{p^E(i,j)} \cdot \beta_T(i,j) \cdot (\alpha_C(i,h-1) \cdot \alpha_{AT}(h,j) \cdot \Pr(T \to CAT)), \\
\widehat{P}_{hl}^E(j,h,l) &= \frac{1}{\alpha_{AT}(h,j)} \cdot (\alpha_A(h,l) \cdot \alpha_T(l+1,j)),
\end{aligned}$$

where

$$\alpha_{AT}(i,j) = \sum\nolimits_{l=(i-1)+\min_{\text{ps}}}^{(j-1)} (\alpha_A(i,l) \cdot \alpha_T(l+1,j))$$

and

$$p^E(i,j) = \beta_T(i,j) \cdot \alpha_T(i,j).$$

Since the probabilities of all mutually exclusive and exhaustive cases sum up to 1, we have $P_0^E(i,j) + P_{ij}^E(i,j) + \sum_{h=(i+1)}^{(j+1)-\min_{\text{ps}}} P_{hj}^E(i,j,h) + \sum_{l=(i-1)+\min_{\text{ps}}}^{(j-1)} P_{il}^E(i,j,l) + \sum_{h=(i+1)}^{j-\min_{\text{ps}}} P_{hl}^E(i,j,h) = 1$, and, under the condition that $P_{hl}^E(i,j,h) > 0$, also $\sum_{l=(h-1)+\min_{\text{ps}}}^{(j-1)} \widehat{P}_{hl}^E(j,h,l) = 1$.

3.2.2 Sampling Probabilities for Other Substructures

In the same way, we can derive equations for computing the needed sampling probabilities for the mutually exclusive and exhaustive cases of any other substructure type, where in any case, the respective equations only depend on the underlying SCFG model and the corresponding inside outside values for the input sequence. The resulting sampling probabilities and their usages directly correspond to those defined and described in [DL03], as in principle the sole difference is that those equations all depend on PFs and free energy values. However, details on all remaining SCFG based sampling probabilities and how they have to be used can be found in Section Sm-II.1.

3.2.3 Sampling Process

A secondary structure for a given RNA sequence $r \in \mathcal{L}_r$ of length n is sampled recursively by starting with the entire RNA sequence R_{1n} and consecutively computing the adjacent substructures (single-stranded regions and paired substructures) of the exterior loop (from left to right), where any paired substructure is completed by successively folding nested substructures. For a formal description of the sampling process (strongly resembling that employed in the Sfold tool), see Algorithms 3 to 6 in Section Sm-II.2.

Note that the sampling process for a secondary structure of a given input sequence r is similar to the traceback algorithm employed in MFE based dynamic programming algorithms. Actually, the main difference is that in those algorithms, base pairings are selected by the minimum energy principle for the fragments R_{ij}, whereas here, base pairs are randomly sampled according to conditional probability distributions for the corresponding fragments, defined by the precomputed inside and outside probabilities and the probabilities of the grammar rules (in contrast to PF approach the where these are derived from precomputed equilibrium PFs and energy values).

We can hence conclude that the considered SCFG based approach and the corresponding PF variant can produce a statistical sample for a given input sequence with similar time and space requirements[8], but the SCFG method can be used with less restrictions (one can allow $\min_{HL} < 3$, non-canonical base pairs and bulge / interior loops of arbitrary length, due to the departure from thermodynamic models). However, when comparing the results of both sampling strategies, significant differences can be observed, as we will see in Section 5.

4 Extension to Structure Prediction

The sampling algorithm sketched in the last section can easily be extended to a prediction algorithm for RNA secondary structures of a single sequence. In principle, after a sample set of possible secondary structures for a given RNA sequence has been constructed, we can derive a corresponding prediction from those (more or less) different candidate structures. Obviously, we can either pick one particular structure from the generated sample as prediction (according to a preliminary defined selection procedure) or we can compute a new structure as predicted folding (according to a preliminary defined construction scheme), where the predicted structure itself must not necessarily be contained in the considered sample. Notably, for the latter variant, there exist elegant ways to incorporate a trade-off parameter $\gamma_{\text{t}-\text{o}}$ in order to provide the user with a mechanism for controlling the *sensitivity* (Sens.) and the *positive predictive value* (PPV)[9] of the predicted foldings. These two measures were introduced in order to quantify the accuracy of RNA secondary structure prediction methods and are usually defined as follows (see e.g. [BBC+00]):

- Sens. is the relative frequency of correctly predicted pairs among all position pairs that are actually paired in a stem of native foldings, whereas

- PPV is defined as the relative frequency of correctly predicted pairs among all position pairs that were predicted to be paired with each other.

Formally, they are given by Sens. $= TP \cdot (TP + FN)^{-1}$ and PPV $= TP \cdot (TP + FP)^{-1}$, where TP is the number of correctly predicted base pairs (*true positives*), FN is the number of base pairs in the native structure that were not predicted (*false negatives*) and FP is the number of incorrectly predicted base pairs (*false positives*).

Note that in [DWB06], the idea of a parameter $\gamma_{\text{t}-\text{o}}$ to control the sensitivity/PPV tradeoff has been used in connection with a dynamic programming optimization scheme. According to its value, the algorithm either tends to predict only those base pairs with rather strong signals for them to belong to the native folding or it is encouraged to predict more pairings even if they might be no part of the native structure. Here, we will show how $\gamma_{\text{t}-\text{o}}$ can be incorporated in connection with sampling algorithms.

4.1 Most Frequent Structure

Since for a sufficiently large sample size, the generated samples are statistically representative, the most frequently observed structure within a given sample set can be assumed to be equal to the most probable folding for the given input sequence (under the considered model, that is according to the corresponding

[8] Both methods can be implemented to run in $\mathcal{O}(n^3)$ time and with $\mathcal{O}(n^2)$ space requirements for a sequence of length n, where a single secondary structure can be drawn in $\mathcal{O}(n^2)$ time.

[9] Note that the positive predictive value is often called *specificity*, like for example in [DWB06], which will be extensively referenced in the sequel.

distribution on the entire ensemble of feasible structures for that sequence). Consequently, for an adequate prediction choice, we simply have to sample a sufficiently large number of possible foldings and choose the most frequently sampled one as prediction. If there are more than one structures sampled with the same highest observed frequency, then the one with the highest probability among all of them should be chosen. This can be considered the standard selection method, as it intuitively yields the "best" sampled structure, which will be denoted by *most frequent (MF)* structure in the sequel.
Obviously, this selection inevitably corresponds to and is thus effectively comparable to the outputs of conventional SCFG based prediction methods for RNA secondary structures from a single sequence. In fact, these methods traditionally determine the most likely parse tree for a given input sequence (under the considered stochastic model) and for structurally unambiguous SCFGs, the most likely parse tree is actually equal to the most probable secondary structure for the given sequence.

4.2 Maximum Expected Accuracy Structures

For so-called *maximum expected accuracy structures* (MEA structures) we employ a rather simple procedure for constructing a particular prediction from a given sample set that uses the trade-off parameter $\gamma_{\mathrm{t-o}}$ as introduced above. Briefly, the MEA structures for a given sequence are the ones among all candidate structures that maximize the number of correctly unpaired positions plus $\gamma_{\mathrm{t-o}}$ times the number of correctly paired positions with respect to the true folding of that sequence. In our case, $\gamma_{\mathrm{t-o}}$ may take on any positive real value and the choice of $\gamma_{\mathrm{t-o}} = 1$ serves as the neutral element with respect to the prediction, i.e. the prediction is neither biased towards a better sensitivity nor to a better PPV. More precisely, $\gamma_{\mathrm{t-o}}$ may take on values in $[0, \infty)$, where for the considered sequence fragment R_{ij}, $1 \leq i, j \leq n$,

- $\gamma_{\mathrm{t-o}} < 1$ restricts the procedure to produce pair $i.j$ only if it is extremely confident,
- $\gamma_{\mathrm{t-o}} = 1$ has no impact on the decision whether $i.j$ should be paired or not,
- $\gamma_{\mathrm{t-o}} > 1$ encourages the algorithm to produce pair $i.j$, even if it is not confident,

that this pair belongs to the native folding.
In [DWB06], this parameter is actually used in the DP algorithm for computing the predicted folding – the MEA structure. More precisely, $\gamma_{\mathrm{t-o}}$ was incorporated into the recursion scheme for calculating the maximum expected accuracy $M_{1,n}$ for an input sequence of length n. In particular, the corresponding DP matrix M is computed according to the following recurrence:

$$M_{i,i} = q_i, \text{ for } 1 \leq i \leq n, \text{ and}$$

$$M_{i,j} = \max \begin{cases} q_i + M_{i+1,j}, \\ M_{i,j-1} + q_j, \\ \gamma_{\mathrm{t-o}} \cdot 2 \cdot p_{i,j} + M_{i+1,j-1}, \\ \max_{i<k<j-1} M_{i,k} + M_{k+1,j}, \end{cases} \quad \text{for } 1 \leq i \leq j-1 \text{ and } 1 \leq j \leq n,$$

where $p_{i,j}$ denotes the probability that i pairs with j and q_i denotes the probability that i remains unpaired. The traceback step of the corresponding DP algorithm can thus be employed to identify the MEA structures of the input sequence according to the given setting of $\gamma_{\mathrm{t-o}}$. If only one MEA structure is recovered in the traceback step, the complete algorithm obviously requires $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ space. Note that for the default setting $\gamma_{\mathrm{t-o}} = 1$, the algorithm only maximizes the expected number of correct (unpaired and paired) positions and is actually identical to the DP technique used in Pfold.
According to [KH03] (supplemental material), a corresponding MEA parser for our sophisticated SCFG and $m_s = 1$ could actually precompute the pairing probabilities $p_{i,j}$, $1 \leq i, j \leq n$, based on the formula[10]

$$\begin{aligned} p_{i,j} =& \beta_A(i,j) \cdot \Pr(A \to \mathbf{(}^{m_s} L \mathbf{)}^{m_s}) \cdot \alpha_L(i+m_s, j-m_s) + \\ & \beta_P(i,j) \cdot \Pr(P \to \mathbf{(}L\mathbf{)}) \cdot \alpha_L(i+1, j-1), \end{aligned} \tag{1}$$

as $A \to \mathbf{(}^{m_s} L \mathbf{)}^{m_s}$ and $P \to \mathbf{(}L\mathbf{)}$ are the only rules that can create paired bases at positions i and j. The respective q_i values, $1 \leq i \leq n$, can then immediately be derived according to

$$q_i = 1 - \sum\nolimits_{j \neq i} p_{i,j}.$$

[10] It should be clear that this formula would only be correct for $m_s = 1$. For choices of $m_s > 1$, however, it would only yield approximate results.

However, contrary to the Pfold [KH03] and CONTRAfold [DWB06] programs, we don't need to derive the $p_{i,j}$'s and q_i's explicitly from our grammar and therefore it is not necessary to find a corresponding formula valid for any possible choice of m_s. In our case, we can easily deduce the needed probabilities from the sample set. This way, we consider the distribution implied by the sample instead of the distribution of the entire structure ensemble of the given input sequence. For a representative sample set however this will make no difference. Accordingly, we will compute the probabilities $p_{i,j}$ by counting the frequencies of all observed base pairs in the particular sample set generated by the sampling algorithm instead of considering the corresponding inside outside values and grammar parameters as done in formula (1). Consequently, the sole difference of our way to compute a MEA structure compared to the CONTRAfold approach lies in the precomputation step, where we will calculate the pairing probabilities according to

$$p_{i,j} = \frac{\text{number of occurrences of pair } i.j \text{ within sample}}{\text{sample size}},$$

such that they depend only on the sampled structures rather than on the entire structure ensemble for the given input sequence. Note that in a clever implementation, $p_{i,j}$ can be determined while constructing the sample. Accordingly, our approach gets rid of the computational overhead needed in cases where (1) is used. Note further that we can make use of this idea in connection with arbitrary sample sets, especially those generated by a PF based approach. MEA structures derived from particular sample sets of candidate foldings for a given setting of the sensitivity/PPV trade-off parameter $\gamma_{\text{t}-\text{o}}$ will be called $\gamma_{\text{t}-\text{o}}$-*MEA structures* (of the respective sample) in the sequel.

4.3 Centroid Structures

The previously proposed selection procedures are especially adequate if one attempts to compare the results to that of other probabilistic prediction methods like the one employed for lightweight SCFGs in [DE04] or those implemented in Pfold and CONTRAfold. This is due to the fact that for a given input sequence, all these algorithms propose only one folding (the one that is assumed to be the "best" under the corresponding model, i.e. the most likely or the MEA structure for the sequence) instead of producing a statistically representative set of candidate structures.

Nevertheless, one benefit of taking on a sampling approach that draws a number of possible foldings from the considered structure ensemble is that we can easily consider alternative schemes for constructing corresponding predictions. Particularly, we can make use of the fact that many (more or less) different secondary structures have been generated by the repeated execution of the sampling procedure and compute a suitable single prediction from the entire sample set. This can be done for example by constructing a particular *consensus* structure like the *centroid* [DCL05] structure of the sample which can be considered as the single structure that best represents the central tendency of the generated sample set.

As the centroid reflects the overall behavior of the structures in the sample, this choice possibly represents an appropriate alternative to the best sampled structure, i.e. the most probable structure according to the considered ensemble distribution implied by the used probabilistic or energy-based[11] approach. Therefore, computing centroids has become custom for applying sampling approaches to single sequence structure prediction. In analogy to the Sfold software we could derive both, ensemble centroids, i. e., centroids computed from the entire set of sampled structures, and cluster centroid, i. e., centroids derived only from a subset of structurally similar samples. However, in order to have a single prediction to be compared to the native folding resp. to the output of other tools we decided to make only use of the first. Furthermore, the ensemble centroid characterizes the central tendency of the entire (representative) sample set and thus is the right choice for what we have in mind, namely for studying the distribution implied by our SCFG.

Formally, the centroid for a given sample set is the structure in the entire structure ensemble that has the minimum total base-pair distance to the structures in the set. It can efficiently be computed as the unique consensus structure formed by all base pairs with a frequency of more than 50%, where the essential matter of fact is that any two base pairs with frequencies $> 50\%$ can not form a pseudoknot. For details, we refer to [DCL05].

In accordance with $\gamma_{\text{t}-\text{o}}$-MEA structures as defined in the last section, we now introduce centroid structures (constructed from sample sets of secondary structures) according to particular settings of the sensitivity/PPV trade-off parameter $\gamma_{\text{t}-\text{o}}$, which will be named $\gamma_{\text{t}-\text{o}}$-*centroids* (of the respective sample) in the sequel. Note that this generalized version of the centroid is very similar to the concept of $\gamma_{\text{t}-\text{o}}$-*centroid estimators* proposed in [HKS+09], which predict the secondary structure maximizing the

[11] Note that the most probable structure is assumed to be (nearly) the MFE structure when sampling is realized via PFs.

expected weighted true predictions of base pairs in the predicted structure on the basis of a particular ensemble distribution for a given RNA sequence (such as for example the Boltzmann distribution or the one implied by a considered SCFG model). In fact, both versions are equivalent to the unique centroid proposed in [DCL05] for $\gamma_{\mathrm{t-o}} = 1$, but the one introduced in [HKS$^+$09] determines the structure that optimizes the *expected numbers* of base pairs of TP, TN[12], FP and FN with respect to the entire ensemble distribution, whereas we only consider the generated sample set for deriving a corresponding $\gamma_{\mathrm{t-o}}$-centroid.

Formally, a $\gamma_{\mathrm{t-o}}$-centroid for a given set of m structures that all have length n is calculated by determining all base pairs $i.j$, $1 \leq i, j \leq n$, which satisfy

$$c_{i,j} = (\text{Number of occurrences of pair } i.j \text{ within sample}) \cdot \gamma_{\mathrm{t-o}} > \frac{m}{2}. \tag{2}$$

These pairs are then used for constructing a corresponding consensus structure, where we have to take care of the fact that the inclusion of any of these pairs into the consensus could eventually result in a pseudoknot or a base triplet which are both prohibited according to our definition of RNA secondary structure.

Therefore, we define the $\gamma_{\mathrm{t-o}}$-centroid as the consensus structure that is formed by successively including base pairs $i.j$ with $c_{i,j} > \frac{m}{2}$ according to their observed frequencies in the sample set (in decreasing order), where $i.j$ is included if and only if it yields a compatible combination (that is, it causes neither a pseudoknot nor a base triplet in the partially formed consensus).

An alternative interpretation of the centroid estimators as introduced in [HKS$^+$09] is the following: The predicted secondary structure maximizes the sum of base-pairing probabilities larger than $\frac{1}{\gamma_{\mathrm{t-o}}+1}$. According to eq. (2) and the strategy just described this is quite similar to our prediction; $c_{i,j} > \frac{m}{2}$ can be rewritten as $\hat{c}_{i,j} :=$ (Number of occurrences of pair $i.j$ within sample)$/m > \frac{1}{2\gamma_{\mathrm{t-o}}}$, where $\hat{c}_{i,j}$ corresponds to a base-pairing probability. By choosing the base pairs according to their decreasing observed frequencies, our strategy to construct a $\gamma_{\mathrm{t-o}}$-centroid aims for maximizing the sum of the $\hat{c}_{i,j} > \frac{1}{2\gamma_{\mathrm{t-o}}}$.

The time complexity for computing one possible $\gamma_{\mathrm{t-o}}$-centroid is bounded by $\mathcal{O}(n^3)$, since any (partially formed) structure of size n can have $\mathcal{O}(n)$ base pairs and we potentially have to check for any of the $\mathcal{O}(n^2)$ possible base pairs whether it can be added to the partially formed centroid or not (i.e. whether it yields a compatible or incompatible combination).

It should be noted that contrary to $\gamma_{\mathrm{t-o}}$-MEA structures, where reasonable values are $\gamma_{\mathrm{t-o}} \in [0, \infty)$, $\gamma_{\mathrm{t-o}}$-centroids by definition might only yield meaningful predictions for $\gamma_{\mathrm{t-o}} \in (\frac{1}{2}, \frac{m}{2})$. Particularly,

- $\gamma_{\mathrm{t-o}} \leq \frac{1}{2}$ leads to $c_{i,j} \leq m \cdot \gamma_{\mathrm{t-o}} \leq \frac{m}{2}$, $1 \leq i, j \leq n$, such that the corresponding centroid contains no base pairs at all,
- $\frac{1}{2} < \gamma_{\mathrm{t-o}} < 1$ results in a unique centroid formed by pairs that have been sampled very often,
- $\gamma_{\mathrm{t-o}} = 1$ produces the unique centroid structure formed by all pairs with a frequency $> 50\%$,
- $1 < \gamma_{\mathrm{t-o}} < \frac{m}{2}$ might produce distinct centroids containing even such pairs that have rarely been sampled,
- $\gamma_{\mathrm{t-o}} > \frac{m}{2}$ implies $c_{i,j} \geq 1 \cdot \gamma_{\mathrm{t-o}} > \frac{m}{2}$ for any pair $i.j$ occurring in the sample, such that the centroid might entirely consist of pairs which have been sampled only once.

However, just like the MF structure, both the $\gamma_{\mathrm{t-o}}$-MEA and $\gamma_{\mathrm{t-o}}$-centroid structures can be calculated from any given set of secondary structures. This means they can not only be employed for obtaining predictions from samples generated with a (sophisticated) SCFG approach, but also from sets of possible foldings created with a corresponding statistical sampling strategy based on PFs. Consequently, this allows for a direct and well-defined comparison of the produced samples with respect to prediction accuracy.

Finally, it might be important to mention that with any of the previously proposed distinct selection processes, the predicted structure can be recovered in $\mathcal{O}(n^3)$ time and with $\mathcal{O}(n^2)$ space requirements, such that the worst-case complexities of the corresponding overall prediction algorithms are equal to those of the respective sampling procedures.

[12] TN is the number of base pairs which were correctly predicted as non-matching (*true negatives*).

5 Evaluation and Discussion

The main objective of this section is to find answers to the two questions from Section 1, and especially to prove or disprove hypothesis H_0. To reach this goal, we will compare our sophisticated SCFG to lightweight SCFGs and corresponding CLLMs for RNA structure prediction. Furthermore, we will discuss the potentials and pitfalls of the corresponding SCFG based sampling method and compare it to the sampling extension of the PF approach as implemented in the Sfold software.

Note that the (purposive) implementation of the statistical sampling strategy sketched in Section 3 (including the corresponding routines for extracting structure predictions as described in Section 4) used for deriving the results of this paper has been incorporated into a web service, which is accessible to the scientific community at `http://wwwagak.cs.uni-kl.de/ProbStatSample`.

5.1 Comparison to Lightweight Grammars and Leading Prediction Methods

In order to see if our sophisticated SCFG can close the performance gap between probabilistic and MFE based approaches and furthermore whether its rich structure and parameter set allows to compensate the powerful scoring schemes of CLLMs (outperforming leading prediction methods) derived from lightweight grammars, we decided to perform a series of cross-validation experiments. Actually, we will compare our grammar to the nine different lightweight SCFGs proposed in [DE04] (to see if its sophisticated design is of any advantage), as well as to the corresponding nine CLLMs and a number of leading prediction methods such as Mfold or ViennaRNA considered in the CONTRAfold paper [DWB06].

It should be mentioned that the nine lightweight SCFGs from [DE04] can be categorized into three groups. First, two structurally ambiguous grammars: G1 is the most simple one (only 5 rules with same left-hand side) and G2 extends it to include base pair stacking parameters. Second, four unambiguous ones: G3 (with 3 intermediates and a total of 8 rules), the smaller G4 (with 2 intermediates and 6 rules), the ultra compact G5 (only one intermediate symbol with 3 alternatives) and G6 (the one utilized in Pfold, with 3 intermediates and 6 rules), where each grammar describes a slightly different class of structures (mainly according to different minimum allowed hairpin lengths). And third, three unambiguous grammars capable of including stacking parameters (and thus prohibiting isolated base pairs): G6s (extension of G6), as well as G7 and G8 (more complex versions of the simple backbones G3 and G4).

Generally, G1 and G5 perform badly, which might be due to the presence of only one nonterminal symbol. Notably, G5 is an extremely bad choice for RNA secondary structure, but a (very) good choice for *covariance models (CMs)*, which are probabilistic models for both, the secondary structure and the primary sequence consensus of an RNA (see, e.g., [RD94]) and are widely used in general approaches to several RNA analysis problems, such as consensus structure prediction, multiple sequence alignment and database similarity searching. The reason, overloading of symbols, leads to this behavior, as for CMs one extends the grammar (by adding rules modeling insertions, deletions and matches), thereby removing the overloading problem (see G5M in [GzS11] for a corresponding specialization of G5).

Nevertheless, since in [DWB06], for each of the nine original lightweight SCFGs from [DE04], an equivalent CLLM has been constructed and two-fold cross-validation procedures[13] have been applied to compare the performances of the respective SCFG and CLLM, we decided to consider the same partition of the structural data set collected in [DWB06] into two folds, such that results reported there can be easily opposed to corresponding ones obtained by our sampling method. Note that this data set contains 151 independent examples of known secondary structures of non-coding RNA from the Rfam database [GJBM$^+$03, GJMM$^+$05], where each independent example has been taken from a different RNA family. It will be denoted by *S-151Rfam database* in the sequel.

For adequate comparisons in case of the lightweight SCFGs and CLLMs, we only considered those principles to derive a prediction from our sample and only corresponding values of $\gamma_{\mathrm{t-o}}$ for which corresponding results are given in [DWB06]. Accordingly, for every structure used for evaluation, we generated a set of 1000 candidate structures[14] with the sampling algorithm and afterwards computed the corresponding MF structure and $\gamma_{\mathrm{t-o}}$-MEA structures, respectively. These predicted foldings were then opposed

[13] In order to perform a k-fold cross-validation, $k \geq 2$, on the basis of a given probabilistic model and a set of real-world data, we first have to partition the data randomly into k approximately equal-sized subsets ("folds"). Then, for any $i \in \{1, \dots, k\}$, we must estimate the model parameters from all objects that are *not* contained in fold i (training set) and validate the results obtained for all objects that actually belong to fold i (benchmark set). The corresponding result of the cross-validation process is then the average of the results derived for the different folds i, $1 \leq i \leq k$.

[14] This sample size has proven to be adequate for most applications, as even for a huge set of possible secondary structures of a given sequence, a sample of only 1000 structures can yield statistical reproducibility of typical sampling statistics, even if samples can be entirely different (see [DL03]).

to the native secondary structure of the molecule (as given in the database) in order to calculate the corresponding sensitivity and PPV, respectively.

Sampling Parameters	MF struct.	
	Sens.	PPV
$\min_{HL} = 1, \min_{\mathrm{hel}} = 1$	0.4433	0.5447
$\min_{HL} = 1, \min_{\mathrm{hel}} = 2$	0.4895	0.5551
$\min_{HL} = 3, \min_{\mathrm{hel}} = 1$	0.4852	0.5948
$\min_{HL} = 3, \min_{\mathrm{hel}} = 2$	0.5171	0.5661

(a) Sensitivity and PPV derived by applying the SCFG based statistical sampling algorithm and choosing the most frequently sampled structure as predicted folding. Notably, all results were computed by two-fold cross-validation procedures, using the same folds of the S-151Rfam database as in [DWB06] and a sample size of 1000 structures.

Grammar	Generative		Discriminative	
	Viterbi		Viterbi	
	Sens.	PPV	Sens.	PPV
G1	0.41	0.27	0.40	0.28
G2	0.53	0.36	0.63	0.48
G3	0.46	0.48	0.45	0.46
G4	0.21	0.17	0.21	0.17
G5	0.03	0.04	0.02	0.03
G6	0.60	0.61	0.61	0.62
G6s	0.60	0.62	0.62	0.63
G7	0.58	0.63	0.58	0.62
G8	0.58	0.60	0.58	0.61

(b) Corresponding results from [DWB06].

Table 1: Comparison of prediction accuracies, obtained by computing the most likely secondary structure for a given sequence by distinct approaches.

Sampling Parameters	MEA struct.		
	Sens.	PPV	$\gamma_{\mathrm{t-o}}$
$\min_{HL} = 1, \min_{\mathrm{hel}} = 1$	0.6029	0.6192	4.0
$\min_{HL} = 1, \min_{\mathrm{hel}} = 2$	0.6325	0.5896	4.0
$\min_{HL} = 3, \min_{\mathrm{hel}} = 1$	0.6090	0.6230	4.0
$\min_{HL} = 3, \min_{\mathrm{hel}} = 2$	0.6311	0.5867	4.0

(a) Sensitivity and PPV derived by applying the SCFG based statistical sampling algorithm and choosing a particular $\gamma_{\mathrm{t-o}}$-MEA structure as predicted folding. Notably, all results were computed by two-fold cross-validation procedures, using the same folds of the S-151Rfam database as in [DWB06] and a sample size of 1000 structures.

Grammar	Generative		Discriminative	
	MEA		MEA	
	Sens.	PPV	Sens.	PPV
G1	0.18	0.11	0.48	0.33
G2	0.53	0.36	0.67	0.64
G3	0.56	0.51	0.54	0.53
G4	0.33	0.23	0.34	0.23
G5	0.06	0.04	0.06	0.04
G6	0.62	0.63	0.62	0.67
G6s	0.62	0.64	0.65	0.65
G7	0.63	0.63	0.63	0.67
G8	0.63	0.62	0.65	0.62

(b) Corresponding results from [DWB06].

Table 2: Comparison of prediction accuracies, obtained by determining a single MEA structure for each given sequence, where the MEA parsing methods are based on the indicated models and $\gamma_{\mathrm{t-o}}$ was adjusted to allow a direct comparison.

The corresponding cross-validation results for the mixed S-151Rfam database are listed in Tables 1 and 2. As we can see from Table 1a, the MF structure predictions obtained by sampling on the basis of our sophisticated SCFG become more accurate when considering the realistic value of $\min_{HL} = 3$. Nevertheless, comparing all results from Table 1 yields the observation that our sophisticated SCFG does not generally outperform any lightweight SCFG and corresponding CLLM, as the most elaborate (generatively or discriminatively trained) grammars G6 to G8 seem to have a greater predictive power when considering the most likely folding of a given input sequence. This might be caused by the fact that the SCFG design underlying the sampling algorithm is too comprehensive to allow for a reliable parameter estimation with respect to the rather sparse but diverse mixed S-151Rfam data set. Table 2 however indicates that when constructing particular $\gamma_{\mathrm{t-o}}$-MEA structures of generated samples, the corresponding prediction results are not significantly less accurate than those obtained by the considered MEA parsing algorithms based on (generatively or discriminatively trained) lightweight grammars. Moreover, there seems to be a slight trade-off between the sensitivity and PPV of the predicted foldings when applying the sophisticated SCFG sampling approach with different values of $\min_{\mathrm{hel}}$, that is when either allowing or prohibiting isolated base pairs (see Table 2a).

For an even more informative comparison of the predictive powers of the distinct lightweight grammar parsing techniques and the sophisticated SCFG based sampling method, the performance has also be

Sampling Parameters	MEA struct.
$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.499491
$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.506602
$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.507454
$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.508762

(a) Estimated AUCs, where the corresponding ROC curves are found by computing one $\gamma_{\text{t}-\text{o}}$-MEA structure for any considered setting of $\gamma_{\text{t}-\text{o}}$ from a particular statistical sample generated by the SCFG based algorithm. Notably, all results were computed by two-fold cross-validation procedures, using the same folds of the S-151Rfam database as in [DWB06] and a sample size of 1000 structures.

Grammar	Generative MEA	Discriminative MEA
G1	0.0392	0.2713
G2	0.3640	0.5797
G3	0.4190	0.4159
G4	0.1361	0.1350
G5	0.0026	0.0031
G6	0.5446	0.5600
G6s	0.5501	0.5642
G7	0.5456	0.5582
G8	0.5464	0.5515

(b) Corresponding results from [DWB06].

Table 3: Comparison of prediction accuracies by means of areas under ROC curves, found by determining a set of MEA structures (for reliable choices of parameter $\gamma_{\text{t}-\text{o}}$) for each given sequence, where the MEA parsing methods are based on distinct models.

measured at several different settings of the $\gamma_{\text{t}-\text{o}}$ parameter. In fact, by determining the (adjusted) sensitivity and PPV for various values of $\gamma_{\text{t}-\text{o}}$, we are able to derive corresponding *receiver operating characteristic (ROC)* curves for the $\gamma_{\text{t}-\text{o}}$-MEA prediction selecting principle (according to the different parameter combinations considered for statistical sampling). Here, we decided to consider any value of $\gamma_{\text{t}-\text{o}} \in \{1.25^k \mid -12 \leq k \leq -1\} \cup \{2^k \mid 0 \leq k \leq 12\}$ in order to obtain appropriate ROC curves. For each curve, the estimated *area under the curve (AUC)* is reported in Table 3a.
Comparing these results to the corresponding values obtained on the basis of the considered lightweight models (Table 3b), we immediately observe that for 5 out of 9 generatively-trained grammars and 4 out of 9 discriminatively-trained grammars, the probabilistic sampling approach (and thus the underlying sophisticated SCFG) yields significantly better results. In all other cases, the sampling variant performs worse, but the corresponding results actually bare no substantial differences with respect to the observed prediction quality.
For a comparison of the predictive accuracy of our sophisticated SCFG sampling approach to several leading probabilistic and physics-based prediction methods, we again considered the S-151Rfam database together with our various strategies to derive a prediction from our samples.
The observed sensitivity and PPV measures are collected in Table 4. First, we observe that accuracies similar to those of Mfold and ViennaRNA can be reached by our SCFG based sampling method when predicting $\gamma_{\text{t}-\text{o}}$-MEA and $\gamma_{\text{t}-\text{o}}$-centroid structures (for adjusted settings of the trade-off parameter $\gamma_{\text{t}-\text{o}}$, respectively), whereas the worst results are obtained when choosing the MF structure as predicted folding (see Table 4a). Furthermore, according to the presented results, our SCFG based sampling approach has been outperformed only by half of the existing probabilistic and energy-based structure prediction methods.

In conclusion, we observe that our sophisticated SCFG cannot significantly improve the predictive power of grammar based methods. Contrarily, the usage of $\gamma_{\text{t}-\text{o}}$-MEA structures as well as $\gamma_{\text{t}-\text{o}}$-centroids introduced in this paper can improve the quality of predictions derived by a sampling approach. The highest values for sensitivity resp. PPV have been observed for $\gamma_{\text{t}-\text{o}}$-centroids ($\gamma_{\text{t}-\text{o}} = 6.0$ resp. $\gamma_{\text{t}-\text{o}} = 1.5$) where we were able to achieve a predictive accuracy close to the one of Mfold and ViennaRNA. However, these observations have been made in connection with a mixed and lean database which might be too small to reliably estimate the rich set of parameters of our grammar. Furthermore, as outlined in the introduction, it might be possible that a sophisticated grammar design is able to capture structural properties (including aspects which are caused by interaction with proteins or by other *non-energetic* details of RNA folding) typical to a single RNA family by the respective parameter values. This possibility – besides other things – will be investigated in the following section. There, we will compare our sampling method to a corresponding physics based approach since that for sure is incapable of adapting to a certain class since its parameters are assumed fixed.

5.2 Comparison of Sample Distributions

Since the considered sampling strategy produces statistically representative sample sets of the complete structure ensemble for a given sequence, we can not only judge the quality of predictions derived from

Sampling Parameters	MF struct.		MEA struct.			Centroid struct.		
	Sens.	PPV	Sens.	PPV	$\gamma_{\text{t-o}}$	Sens.	PPV	$\gamma_{\text{t-o}}$
$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.4433	0.5447	0.6342	0.5842	6.0	0.6522	0.5612	6.0
			0.5083	0.6808	2.0	0.4387	0.7180	1.5
$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.4894	0.5551	0.6546	0.5593	6.0	**0.6624**	0.5354	6.0
			0.4980	0.6850	1.5	0.4801	0.6977	1.5
$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.4852	0.5948	0.6348	0.5826	6.0	0.6464	0.5616	6.0
			0.4627	0.7044	1.5	0.4487	**0.7241**	1.5
$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.5171	0.5661	0.6502	0.5568	6.0	0.6411	0.5700	4.0
			0.4342	0.7228	1.0	0.4917	0.7103	1.5

(a) Sensitivity and PPV derived by applying the SCFG based statistical sampling algorithm and selecting the predicted folding according to any of the described schemes. Notably, all results were computed by two-fold cross-validation procedures, using the same folds of the S-151Rfam database as in [DWB06] and a sample size of 1000 structures.

Method	References	Sens.	PPV	$\gamma_{\text{t-o}}$
CONTRAfold	[DWB06]	0.7377	0.6686	6.0
Mfold v3.2	[Zuk89, Zuk03]	0.6943	0.6063	–
ViennaRNA v1.6	[HFS+94, Hof03]	0.6877	0.5922	–
PKNOTS v1.05	[RE99]	0.6030	0.5269	–
ILM	[RSZ04]	0.5330	0.4098	–
CONTRAfold	[DWB06]	0.5540	0.7920	0.75
Pfold v3.2	[KH99, KH03]	0.4906	0.7535	–

(b) Accuracies of other methods, as reported in [DWB06].

Table 4: Comparison of the sophisticated SCFG sampling approach to leading secondary structure prediction methods (that are not based on sampling).

a particular sample, but also the quality of the generated sample as it. In this section, we will compare the sample distribution implied by our sophisticated SCFG to the one induced by the PF based sampling method as implemented in the Sfold software. For that purpose, we will consider probability profiles as well as (and most interestingly from the perspective of biologists) a number of different comparisons on the basis of *abstract shapes* as introduced in [GVR04, SVR+06, JRG08]. Abstract shapes are morphic images of secondary structures (which in the sequel will be assumed the level 0 shape), where each shape comprises a class of similar foldings. The motivation behind this concept is that the predicted set of suboptimal foldings for a given sequence (as computed by modern secondary structure prediction tools) usually contains lots of similar structures that obey to (almost) identical structural properties, but for biologists only those with significant structural differences are of interest.

Briefly, there are five shape types for five different levels of abstraction. Two of them, namely type 1 and type 5 (also called π' and π shapes, respectively), were formally defined by a tree morphism in [GVR04]. All five different shape levels were first introduced and informally described in [SVR+06] and were later redefined (informally) in [JRG08]. Common to all levels is their abstraction from loop and stem lengths, while generally retaining nesting and adjacency of helices, but disregarding their size and concrete position in the primary structure. In the most accurate shape type (type 1), all structural components (except hairpin loops) contribute to the shape representation. The succeeding shape types are supposed to gradually increase abstraction by disregarding certain unpaired regions or combining nested helices. For the renewed shape abstraction types as described in [JRG08], it has been proven that this is the case indeed [NS09].

Finally, before we start our examinations, it should be mentioned that in order to derive all results for the particular applications that will follow throughout this section, we have implemented our own version of Sfold's sampling procedure as described in [DL03]. For this implementation, we decided to use the common thermodynamic parameters from Mathews et al. [MSZT99], which were also used for version 3.0 of the Mfold software [Zuk03].

5.2.1 RNA Data

For the previously mentioned reasons, we decided to no only consider the mixed S-151Rfam database for our subsequent comparisons, but also use several other databases that contain more structures having more similar shapes. In particular, we took the tRNA database from [SHB+98], where we filtered out

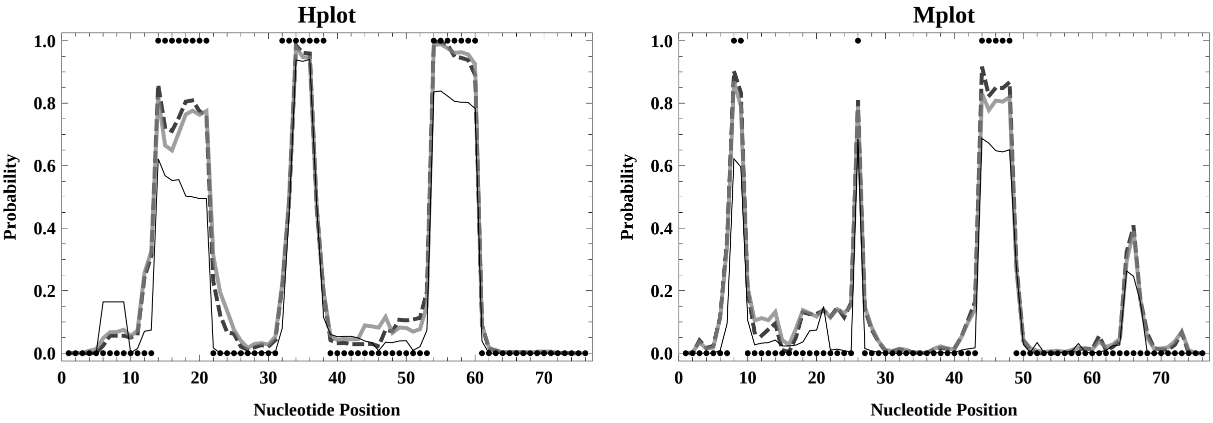


Figure 1: Loop profiles for *E.coli* tRNA^{Ala}. Hplot and Mplot display the probability that an unpaired base lies in a hairpin and multibranched loop, respectively. Results for the PF approach (for $\max_{BL} = 30$) are displayed by the thin black lines. For the SCFG approach, we chose $\min_{\text{hel}} = 1$ (thick gray lines) and $\min_{\text{hel}} = 2$ (thick dashed darker gray lines), combined with $\min_{HL} = 3$, respectively. The corresponding probabilities for the correct structure of *E.coli* tRNA^{Ala} are also displayed (by black points).

all sequences with unidentified bases, yielding a total of 2163 distinct tRNA structures (having lengths in $[64, 93]$ and an average length of 76). Additionally, we created another set of 1149 distinct sequences (with lengths in $[102, 135]$ and about 119 on average), retrieved from a 5S rRNA database [SBEB02]. These data sets of tRNA and 5S rRNA structures, along with the mixed S-151Rfam set, will be the basis for the following studies.

5.2.2 Probability Profiling for Specific Loop Types

A representative sample of all possible secondary structures for a given RNA sequence can be used to derive estimates for the probability (conditioned to the sequence) of any structural motif to show up at the different sequence positions. For example, *probability profiling* of unpaired bases in RNA secondary structure becomes possible, i.e. paired and unpaired bases are delineated on statistic grounds derived from the sample set. In detail, for probability profiling, the unpaired bases can either be delineated regardless of the type of loop (like hairpin, bulges and so so) in which they occur. Or, by keeping track of the loop type for unpaired bases, an extension that accounts for the different types of loops is possible. For each nucleotide position i, $1 \leq i \leq n$, of a given sequence of length n, one computes the probabilities that i is an unpaired base within a specific loop type. These probabilities are given by the observed frequency in a sample set of secondary structures for the given sequence.

For a first comparison of the two different sample distributions, we decided to consider the corresponding probability profiles for *Escherichia coli* tRNA^{Ala}. Using a sample size of 1000 structures, we obtain the ten profile plots shown in Figure 2 of Section Sm-III. The potentially most interesting ones are presented in Figure 1 which obviously exhibit the cloverleaf structure of tRNAs.

All these profiles show that the (statistically representative and reproducible) samples generated by the SCFG approach are significantly more accurate than those obtained with the PF approach. Moreover, considering the results for this tRNA example under the assumption of $\min_{HL} = 3$ (which is always implicitly chosen for the PF approach), we see that the quality of sample sets can be further improved by increasing the minimum allowed helix size $\min_{\text{hel}}$. Moreover, under the assumption of the less realistic minimum hairpin loop size $\min_{HL} = 1$, the generated results are qualitatively not as good as those for $\min_{HL} = 3$ (see Figure 2 in Section Sm-III).

5.2.3 The Problem of Overfitting and the Lack of Generalization

In this section we will address two possible issues of our sophisticated grammar in connection with this study: the problem of overfitting and the lack of generalization. With respect to the latter, it might not be surprising to some readers that the profile plots for *Escherichia coli* tRNA^{Ala} presented in Figure 1 indicate an accuracy gain of the probabilistic SCFG approach over the physics-based PF variant for the following reasons: First, it seems inevitable that a sophisticated stochastic model that is trained on trusted tRNAs only produces the typical tRNA cloverleaf shape more often than an alternative variant that is not tailored to a specific structure class but only relies on free energy, such that the SCFG based profiles should inherently show the cloverleaf structure more explicitly. Additionally, it is known that SCFG based approaches work well for short RNA types whose molecules imply a low structural variety, whereas the standard thermodynamic model for RNA secondary structures might perform poorly on some tRNAs [RCM99].

For these reasons, it might be assumed that the higher accuracy reached by the probabilistic sampling approach could be an artefact caused by a lack of generalization of the underlying SCFG model. To show that this is not the case, we performed a series of experiments based on (more and less arbitrary) random sequences. In principle, for any chosen value of $\mathrm{min}_{\mathrm{hel}} \in \{0, \dots, 7\}$, we generated a set of random RNA sequences in the following way: for a considered sequence length n, we randomly created a number of (not necessarily distinct) secondary structures of size n having the cloverleaf shape, where all four helices (the stem and the three adjacent helices of the multiloop) are formed by exactly $\mathrm{min}_{\mathrm{hel}}$ consecutive base pairs. For any of these cloverleaf structures, we then generated a corresponding sequence by randomly drawing canonical base pairs for the helical regions and arbitrary unpaired bases for the single-strands.

Approach	$\mathrm{min}_{\mathrm{hel}}$	$\mathrm{num}_{\mathrm{d}}$	c_{d}	c_{MF}	c_{CL}	$\mathrm{num}_{\mathrm{MF}}$
PF	0	36	8333.33	94085	3331	6
	1	34	8823.53	87785	5338	6
	2	35	8571.43	96083	2745	6
	3	37	8108.11	95332	4492	6
	4	30	10000.	107881	9967	6
	5	29	10344.8	111716	20875	3
	6	33	9090.91	102788	49733	2
	7	27	11111.1	94859	94859	0
SCFG	0	858	349.65	26341	14114	5
	1	916	327.511	22643	15596	4
	2	915	327.869	21258	13912	4
	3	895	335.196	20175	16207	2
	4	914	328.228	19828	17784	2
	5	844	355.45	20560	20560	0
	6	747	401.606	34753	34753	0
	7	658	455.927	59644	59644	0

Table 5: Results derived from random data sets, where $\mathrm{min}_{\mathrm{hel}}$ has been used for generating random sequences with corresponding (more or less strong) signals towards a cloverleaf structure. $\mathrm{num}_{\mathrm{d}}$ denotes the number of distinct shapes in all samples and c_{d} the average count of one of these distinct shapes. Furthermore, c_{MF} and c_{CL} represent the count of the most frequent and cloverleaf shape in all samples, whereas $\mathrm{num}_{\mathrm{MF}}$ denotes the number of distinct shapes that are observed more frequently than the cloverleaf. For any setting of $\mathrm{min}_{\mathrm{hel}}$, all tabulated values were computed from a corresponding random data set of cardinality 300 (containing 10 random sequences for any length $n \in \{64, \dots, 93\}$ according to the length range observed from our tRNA database), respectively. A sample size of 1000 structures and $\max_{BL} = 30$ has been chosen for either approach.

Obviously, regardless of the applied sampling approach, the signal towards generating the actual cloverleaf structure should get stronger with increasing value of $\mathrm{min}_{\mathrm{hel}}$ and for $\mathrm{min}_{\mathrm{hel}} = 0$, there is absolutely no signal towards the cloverleaf shape, since the corresponding structures have been generated completely at random (by drawing all nucleotides in the sequence independently). As we can see from Table 5 (where the corresponding results have been derived for the most abstract shape level 5), both sampling approaches tend to primarily generating cloverleaf structures if the signals are strong enough, but other shapes are sampled more often if the signal towards cloverleaf is low or does actually not exist. Basically, the SCFG based variant seems to react faster to such signals (by preferring the cloverleaf shape over others more notably already for rather low signals compared to the PF method). However, since for actual random sequences, the typical cloverleaf shape of tRNAs is neither sampled all the time nor significantly more often than any other shape (among a vast number of distinct ones that are observed), there is no reason to believe that the accuracy of the SCFG based sampling strategy (at least for tRNAs) is due to a lack of generalization (or the other way round is due to a model tailored to a certain shape). Since we most likely observe such effects in connection with tRNA and its invariant cloverleaf shape, we skipped similar investigations for the other cases.

To see if overfitting is not a problem for our experiments, i.e. to see if our data sets are rich enough to reliably derive the parameters of our grammar, we performed the following experiments: For each

$\mathbb{V}[\cdot]$	tRNA	5S rRNA	S-151 Rfam
p_1	0	0	0
p_2	5.747×10^{-8}	2.232×10^{-6}	1.613×10^{-5}
p_3	1.223×10^{-7}	6.635×10^{-6}	8.673×10^{-6}
p_4	3.745×10^{-8}	2.718×10^{-6}	1.012×10^{-5}
p_5	9.954×10^{-7}	3.437×10^{-6}	1.983×10^{-5}
p_6	9.579×10^{-7}	1.697×10^{-6}	4.120×10^{-5}
p_7	8.853×10^{-6}	2.849×10^{-5}	7.766×10^{-6}
p_8	8.853×10^{-6}	2.849×10^{-5}	7.766×10^{-6}
p_9	0	0	0
p_{10}	0	0	0
p_{11}	4.541×10^{-9}	1.385×10^{-9}	1.362×10^{-6}
p_{12}	2.645×10^{-8}	8.330×10^{-8}	2.264×10^{-6}
p_{13}	8.500×10^{-9}	6.674×10^{-8}	4.074×10^{-6}
p_{14}	6.762×10^{-10}	3.464×10^{-10}	3.270×10^{-7}
p_{15}	0	0	0
p_{16}	1.234×10^{-8}	7.211×10^{-9}	5.812×10^{-6}
p_{17}	1.234×10^{-8}	7.211×10^{-9}	5.812×10^{-6}
p_{18}	0	1.152×10^{-6}	5.352×10^{-5}
p_{19}	0	3.919×10^{-7}	2.957×10^{-5}
p_{20}	0	4.502×10^{-7}	8.094×10^{-5}
p_{21}	2.695×10^{-3}	2.997×10^{-8}	4.429×10^{-5}
p_{22}	2.695×10^{-3}	2.997×10^{-8}	4.429×10^{-5}
p_{23}	0	0	0
p_{24}	0	0	0
p_{25}	0	0	1.333×10^{-4}
p_{26}	0	0	1.333×10^{-4}
p_{27}	4.052×10^{-7}	1.561×10^{-7}	1.347×10^{-4}
p_{28}	4.052×10^{-7}	1.561×10^{-7}	1.347×10^{-4}
p_{29}	0	0	0

Table 6: Truncated variances of parameters derived from 100 iterations of training our grammar on random subsets of the original training data.

RNA type considered and $\min_{\text{hel}} = 2$, $\min_{HL} = 3$ we selected a random 90% portion of the original database (the resulting sample size equals that of the training sets used for our k-fold cross-validation experiments) and re-estimated the probabilities of all the grammar rules. This process was iterated 100 times, resulting in a sample of 100 parameter sets. Finally, for each parameter we determined its variance along this sample of size 100. The corresponding values are presented in Table 6. Note that the variances 0 in most cases result for intermediate symbols without alternatives; for whose productions a probability of 1 is predetermined. However, all the other variances are rather small too and we can conclude that overfitting is no issue in connection with our sophisticated grammar and the training sets used.

5.2.4 Prediction Accuracy – Sensitivity and PPV

To compare the quality of predictions derived from samples generated by the PF approach to those implied by our SCFG, we again performed two-fold cross-validations based on the mixed S-151Rfam data set. Furthermore, we partitioned the more comprehensive tRNA and 5S rRNA databases into 10 approximately equal-sized folds and derived corresponding 10-fold cross-validations results, respectively. The determined sensitivity and PPV measures are collected in Tables 7 to 9. Note that for any sequence, we predicted one structure according to each of the principles introduced in Section 4, where for the sake of completeness we considered the default choice $\gamma_{\text{t-o}} = 1$ for MEA and centroid structures, as well as varying values for $\gamma_{\text{t-o}}$ (the same ones as considered above) to obtain AUC values (plots of some of the respective ROC curves can be found in Figures 3, 4 and 5 of Section Sm-III). Obviously, the provided AUC values allow for a reliable comparison of the accuracies that can be reached by either sampling approach when calculating $\gamma_{\text{t-o}}$-MEA and $\gamma_{\text{t-o}}$-centroid structures for the produced samples.

Let us first consider the results presented in Table 7. Here, we observe that for the low invariant tRNAs, the accuracy of predictions computed by statistical sampling methods can be significantly improved when using the SCFG approach. Moreover, the quality of predictions can be further improved by considering the realistic value of $\min_{HL} = 3$ (also implicitly chosen for the PF approach) instead of the unrealistic choice $\min_{HL} = 1$. However, it seems that increasing the value of parameter $\min_{\text{hel}}$ does not have a mentionable impact on the resulting prediction accuracy.

According to Table 8, the predictions for 5S rRNAs are less accurate than for tRNAs. In detail, for 5S RNAs the predictive accuracy as measured by sensitivity and PPV is slightly higher for the PF approach when selecting the most frequently sampled structure as prediction. By constructing a MEA

Approach	Parameters	MF struct.		MEA struct.		Centroid	
		Sens.	PPV	Sens.	PPV	Sens.	PPV
PF	$\max_{BL} = 30$	0.6565	0.5890	0.6434	0.6035	0.6159	0.6344
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.7791	0.8445	0.7324	0.8939	0.6754	0.9158
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.8004	0.8457	0.7685	0.8878	0.7113	0.9123
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.8545	0.8517	0.7848	0.9021	0.7304	0.9213
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.8677	0.8593	0.8182	0.8953	0.7713	0.9168

(a) Sensitivity and PPV (computed by 10-fold cross-validation procedures, using sample size 1000).

Approach	Parameters	MEA struct.	Centroid
PF	$\max_{BL} = 30$	0.482435	0.526743
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.828522	0.833894
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.830787	0.839843
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.855406	0.861640
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.857251	0.867135

(b) AUC values (computed by 10-fold cross-validation procedures, using sample size 1000).

Table 7: Prediction results for our tRNA database.

Approach	Parameters	MF struct.		MEA struct.		Centroid	
		Sens.	PPV	Sens.	PPV	Sens.	PPV
PF	$\max_{BL} = 30$	0.5897	0.5806	0.6015	0.6191	0.5789	0.6508
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.4251	0.5362	0.3403	0.6967	0.2689	0.8044
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.4542	0.5435	0.3638	0.6901	0.2727	0.8069
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.4728	0.5290	0.3544	0.7033	0.2764	0.8091
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.5167	0.5577	0.3860	0.7010	0.2846	0.8140

(a) Sensitivity and PPV (computed by 10-fold cross-validation procedures, using sample size 1000).

Approach	Parameters	MEA struct.	Centroid
PF	$\max_{BL} = 30$	0.481019	0.520171
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.409278	0.408549
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.417286	0.418584
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.419116	0.417095
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.433954	0.431642

(b) AUC values (computed by 10-fold cross-validation procedures, using sample size 1000).

Table 8: Prediction results for our 5S rRNA database.

structure and especially the unique centroid structure, however, we observe significant differences between both sensitivity and PPV obtained by either sampling approach. The corresponding AUCs confirm the advantages of the PF approach on these data. Furthermore, the case $\gamma_{\text{t-o}} = 1$ implies that base pairings of the native foldings generally occur less frequently in samples generated by the SCFG based algorithm (FN is greater), but the sampled pairs are more often correct (FP is smaller). Considering the unique centroid predictions, this means that the SCFG method rarely samples incorrect pairings (otherwise, those would be part of the prediction), while pairs which are sampled with a high frequency typically are native ones. This decreased precision may be implied by the comparably high structural diversity of 5S rRNAs and the corresponding reduced ability of our SCFG model to capture typical structural features of the considered family within its parameters.
Last but not least, similar results can be observed for the S-151Rfam data set in connection with the default choice $\gamma_{\text{t-o}} = 1$, as shown in Table 9a. In fact, the performance gap between the two different sampling approaches remains quite the same as for our 5S rRNA database, although this mixed data set is less comprehensive and contains structures that not only belong to distinct RNA types but also partially contained pseudoknots that had to be removed, such that this S-151Rfam set might not be considered a high-quality training basis. In contrast to the 5S rRNAs however, considering the AUC values of Table 9b reveals slight advantages of our SCFG over PFs.

Approach	Parameters	MF struct.		MEA struct.		Centroid	
		Sens.	PPV	Sens.	PPV	Sens.	PPV
PF	$\max_{BL} = 30$	0.6652	0.5188	0.6633	0.5450	0.6437	0.5799
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.4433	0.5447	0.3815	0.7386	0.3235	0.7749
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.4894	0.5551	0.4263	0.7181	0.3474	0.7743
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.4852	0.5948	0.3935	0.7426	0.3352	0.7825
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.5171	0.5661	0.4342	0.7228	0.3588	0.7683

(a) Sensitivity and PPV (computed by two-fold cross-validation procedures, using the same folds as in [DWB06] and sample size 1000).

Approach	Parameters	MEA struct.	Centroid
PF	$\max_{BL} = 30$	0.450688	0.497350
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.499491	0.507125
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.506602	0.509403
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.507454	0.512327
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.508762	0.514958

(b) AUC values (computed by two-fold cross-validation procedures, using the same folds as in [DWB06] and sample size 1000).

Table 9: Prediction results for the mixed S-151Rfam database.

In conclusion, we have three different scenarios for the three different data sets: for tRNAs our SCFG performs best for fix and varying $\gamma_{\text{t-o}}$, for 5S rRNA the PF approach is superior in both cases and for the S-151 Rfam data set the SCFG is beaten by the PF approach for $\gamma_{\text{t-o}} = 1$ while the SCFG gives rise to better AUCs.

5.2.5 Sampling Quality – Specific Values Related to Shapes

Note that the previously considered measures for assessing the accuracy of secondary structure predictions (sensitivity and PPV) depend only on the numbers of correctly and incorrectly predicted base pairs (compared to the native structure). From the perspective of biologists, however, it is usually much more important to get information on the correct structural properties (described by the corresponding abstract shapes) of the native folding than to obtain high sensitivity and PPV when using computational prediction methods.

Therefore, in order to further investigate the sampling quality, we decided to consider the following specific values related to the shapes of sampled structures:

- Frequency of prediction of correct structure (CSP_{freq}): In how many cases is the predicted secondary structure (or its shape) equal to the correct structure (or the correct shape)?
- Frequency of correct shape occurring in a sample (CSO_{freq}): In how many cases can the correct shape (on different levels) be found in the generated sample set?
- Number of occurrences of correct shape in a sample (CS_{num}): How many times can the correct shape be found in the generated sample set?
- Number of different shapes in a sample (DS_{num}): How many different secondary structures (or shapes) can be found in the generated sample set?

To compute the desired values, we considered the predicted structures and the corresponding sample sets that were derived for the calculation of the sensitivity and PPV measures in the last section (Tables 7a, 8a and 9a). The respective results are collected in Tables 13 to 18 in Section Sm-III. Some of the most interesting ones are displayed in Tables 10 to 12.

Comparing the corresponding values, we immediately observe that for our tRNA and 5S rRNA databases, the predicted shapes are in almost all cases significantly more often equal to the correct ones when using the SCFG based sampling strategy instead of the PF alternative. This means given rich and explicit training data, the frequency of correct structure predictions (CSP_{freq}) is basically higher when relying on the ensemble distribution induced by our sophisticated SCFG. Moreover, the samples generated with the SCFG method generally contain the correct shapes considerably more often than those obtained with the corresponding PF algorithm and are thus more accurate as regards the frequency of correct structure occurrences (CSO_{freq}).

Value	Approach	Parameters	Shape Level					
			0	1	2	3	4	5
CSP_{freq}	PF	$\text{max}_{BL} = 30$	0.0633	0.1216	0.2071	0.2117	0.2639	0.3694
(MF struct.)	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 1$	0.2450	0.4448	0.6417	0.6417	0.6422	0.7356
CSP_{freq}	PF	$\text{max}_{BL} = 30$	0.0416	0.1049	0.1923	0.1960	0.2496	0.3559
(MEA struct.)	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	0.1008	0.2917	0.5525	0.5525	0.5543	0.6241
CSP_{freq}	PF	$\text{max}_{BL} = 30$	0.0264	0.0800	0.1595	0.1627	0.1932	0.2677
(Centroid)	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	0.0758	0.2150	0.4563	0.4563	0.4568	0.5003
CSO_{freq}	PF	$\text{max}_{BL} = 30$	0.5196	0.6740	0.8160	0.8239	0.8798	0.9556
	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 1$	0.7148	0.9459	0.9875	0.9880	0.9885	0.9991
CS_{num}	PF	$\text{max}_{BL} = 30$	21.073	58.200	136.67	140.63	205.54	328.56
	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	34.898	173.73	513.05	513.06	513.08	595.26
DS_{num}	PF	$\text{max}_{BL} = 30$	355.32	130.22	81.796	33.125	22.585	4.8848
	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	592.84	103.04	18.921	18.921	18.921	12.053

Table 10: Results related to the shapes of selected predictions and sampled structures, obtained from our tRNA database (by 10-fold cross-validation procedures, using sample size 1000).

Value	Approach	Parameters	Shape Level					
			0	1	2	3	4	5
CSP_{freq}	PF	$\text{max}_{BL} = 30$	0.0000	0.0009	0.0078	0.0513	0.0261	0.6353
(MF struct.)	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	0.0009	0.0096	0.0244	0.0609	0.1027	0.8207
CSP_{freq}	PF	$\text{max}_{BL} = 30$	0.0000	0.0052	0.0139	0.0835	0.0696	0.6640
(MEA struct.)	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	0.0000	0.0009	0.0009	0.0035	0.0557	0.5387
CSP_{freq}	PF	$\text{max}_{BL} = 30$	0.0000	0.0026	0.0104	0.0775	0.0731	0.7214
(Centroid)	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	0.0000	0.0000	0.0000	0.0009	0.0139	0.1549
CSO_{freq}	PF	$\text{max}_{BL} = 30$	0.0009	0.1662	0.3063	0.7580	0.6883	0.9817
	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	0.0026	0.4509	0.6372	0.9904	0.9974	0.9991
CS_{num}	PF	$\text{max}_{BL} = 30$	0.0009	0.7571	3.4207	36.641	30.288	600.35
	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	0.0026	1.3795	3.1949	36.673	71.080	609.58
DS_{num}	PF	$\text{max}_{BL} = 30$	710.75	333.72	237.71	93.335	63.661	7.0951
	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	999.68	885.81	762.67	239.28	123.91	13.558

Table 11: Results related to the shapes of selected predictions and sampled structures, obtained from our 5S rRNA database (by 10-fold cross-validation procedures, using sample size 1000).

Value	Approach	Parameters	Shape Level					
			0	1	2	3	4	5
CSP_{freq}	PF	$\text{max}_{BL} = 30$	0.0661	0.1255	0.1586	0.2050	0.2183	0.4834
(MF struct.)	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	0.0530	0.1258	0.1522	0.1788	0.1985	0.4240
CSP_{freq}	PF	$\text{max}_{BL} = 30$	0.0660	0.1123	0.1453	0.1984	0.2051	0.4902
(MEA struct.)	SCFG	$\text{min}_{HL} = 1, \text{min}_{\text{hel}} = 2$	0.0264	0.1193	0.1391	0.1523	0.1789	0.4239
CSP_{freq}	PF	$\text{max}_{BL} = 30$	0.0793	0.1321	0.1653	0.1917	0.2449	0.5100
(Centroid)	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	0.0197	0.0927	0.1125	0.1390	0.1391	0.3577
CSO_{freq}	PF	$\text{max}_{BL} = 30$	0.3638	0.4433	0.4766	0.5231	0.6488	0.7947
	SCFG	$\text{min}_{HL} = 1, \text{min}_{\text{hel}} = 2$	0.2717	0.5630	0.6158	0.7284	0.8079	0.9605
CS_{num}	PF	$\text{max}_{BL} = 30$	40.390	88.886	121.55	158.32	195.83	453.58
	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	15.059	63.707	83.965	125.82	142.99	391.39
DS_{num}	PF	$\text{max}_{BL} = 30$	540.74	304.36	255.40	150.89	117.24	18.795
	SCFG	$\text{min}_{HL} = 3, \text{min}_{\text{hel}} = 2$	840.03	522.53	452.04	307.61	273.92	77.536

Table 12: Results related to the shapes of selected predictions and sampled structures, obtained from the S-151Rfam database (by 2-fold cross-validation procedures, using sample size 1000).

However, having only a lean training set of mixed RNAs like the S-151Rfam database at hand, then the energy-based sampling approach seems to outperform its probabilistic counterpart, at least with respect to shape prediction.

Furthermore, as regards tRNAs and 5S rRNAs, the observed averaged number of correct shapes in a sample set (CS_{num}) is greater when using the SCFG approach, whereas for the S-151Rfam set of mixed structural RNAs, an arbitrary sample obviously contains more instances of the correct shape when using the PF variant. For 5Sr RNAs, this observation especially holds for the two most interesting shape types (the most accurate shape type 1 and the most abstract type 5) with the realistic parameter choice $\text{min}_{\text{hel}} = 2$ for the SCFG strategy (see Table 16 of Section Sm-III). Finally, the observed averaged number of different shapes in a sample (DS_{num}) is in most cases significantly larger for the SCFG based sampling method[15]. This actually means that samples generated according to the distribution induced by a sophisticated SCFG design imply a greater diversity of candidate structures for a given input sequence than corresponding Boltzmann samples.

Consequently, the SCFG based statistical sampling approach evaluated within this article effectively overcomes the main pitfall of MFE based methods addressed in the introduction, namely that the predicted set of suboptimal foldings for a given sequence usually contains mostly structures without fundamental differences. However, there is neither clear evidence that the distribution induced by a sophisticated SCFG generally yields more realistic results than a corresponding energy-based Boltzmann distribution, nor the other way round. In fact, this seems to strongly depend on the RNA type of the given sequence, and most importantly on the quality of a corresponding training set and on the performance of the thermodynamic model on such RNAs. Altogether, we conclude that fundamental differences might be expected between Boltzmann samples and corresponding statistical sample sets obtained by a sophisticated SCFG approach, which eventually disproves hypothesis H_0 proposed in Section 1.

6 Conclusion and Future Work

In this work, we evaluated a sophisticated SCFG that mirrors the standard thermodynamic model applied in modern physics-based RNA secondary structure prediction methods. Particularly, this rather complex SCFG represents an exact probabilistic counterpart to the energy model employed for calculating the needed PFs for the sampling strategy implemented in the Sfold program [DL03, DCL04], which has become a widely used tool for RNA structure prediction based on statistical characterizations of the thermodynamic ensemble of suboptimal foldings. We effectively used that elaborate SCFG design as foundation of a corresponding sampling method that samples possible foldings of a given RNA molecule rigorously from the induced probability distribution. In principle, that SCFG based sampling strategy produces a statistically representative sample of secondary structures for a given input sequence in proportion to the distribution on the entire ensemble of feasible foldings, which is implied by the learned grammar parameters. Thus, this sampling method represents a probabilistic counterpart to the energy-based PF variant of Sfold, where structures are sampled in proportion to their Boltzmann weights, guaranteeing a statistical representation of the Boltzmann-weighted ensemble.

By comprehensive comparisons, we showed that incorporating only additional information obtained from databases of trusted RNA sequences with annotated secondary structures (SCFG variant) instead of the recent thermodynamic parameters for RNA secondary structure (PF variant) into a statistical sampling algorithm results in significant differences with respect to both predictive accuracy and overall quality of generated sample sets. Actually, we can draw the conclusion that the ensemble distribution induced by the considered sophisticated SCFG is less centered than the corresponding Boltzmann distribution of possible structures. This effectively yields more variability during the sampling process and consequently reduces the problem of getting stuck in local optima (which is inevitably inherited from optimization algorithms), resulting in a more diverse sample set that might also contain structures which are fundamentally different to the most probable ones. Thus, the discussed probabilistic sampling approach may be used to address exactly the critical features of deterministic structure prediction methods and hence eventually realizes the intentions related to statistical sampling techniques towards RNA structure prediction.

However, there is still room for improvement. For example, when using a so-called *length-dependent* stochastic context-free grammar (LSCFG) as recently introduced in [WN10] to model RNA secondary structures, it is very likely that the performance of the probabilistic sampling strategy employed in this work can be enhanced, in terms of both accuracy of predictions and overall sampling quality.

Finally, note that despite the potential major quality improvement of the SCFG variant over the PF approach for certain RNA types, the worst-case time complexity and memory requirement for the construction of a statistically representative and reproducible sample for a given sequence are actually the same. According to these aspects, the SCFG approach that has been evaluated within this article may

[15] Note that in the few cases where the PF approach yields more different shapes, we generally further restricted the possible structures by prohibiting isolated base pairs ($\text{min}_{\text{hel}} > 1$), which are in fact allowed in PF calculations.

inspire the development of new high quality (sampling) algorithms, for example for RNA structures with pseudoknots or RNA-RNA interactions, due to the following reasons: Despite the fact that RNA structure prediction including pseudoknots based on thermodynamics is $\mathcal{NP}$-hard, some MFE based algorithms have been developed to include certain types of pseudoknots [RE99, RG04], but due to their high time and space complexities, these particular algorithms are not applicable for long sequences. Moreover, the PF algorithm [McC90] has been extended to include a class of pseudoknots [DP03, DP04], such that a sampling extension could also be developed for structures including pseudoknots. However, one of the main problems with these approaches is their dependence on the thermodynamic parameters and energy functions which limits the performance accuracies in very significant ways, since there exists little knowledge on the thermodynamic behavior of pseudoknotted structures. Nevertheless, it is known how to model RNA structures with pseudoknots (and also RNA-RNA interactions) by special more powerful grammar models, such that one does not have to face the problem that no appropriate energy parameters are available. Thus, by completely abstracting from thermodynamics and considering only typical structural information obtained by training a convenient grammar on structural databases, one might be able to generalize the sampling strategy discussed in this work to an algorithm for predicting pseudoknotted RNA secondary structures (or RNA-RNA interactions).

Acknowledgements

The authors wish to thank two anonymous reviewers for their careful and helpful remarks and suggestions made for a previous version of this article.

References

[BBC+00] P. Baldi, S. Brunak, Y. Chauvin, C. A. Andersen, and H. Nielsen. Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16(5):412–424, 2000.

[CPG83] R. Chaudhuri, S. Pham, and O. N. Garcia. Solution to an open problem on probabilistic grammars. *IEEE Trans. on Computers*, C-32(8):748–750, 1983.

[DCL04] Y. Ding, C. Y. Chan, and C. E. Lawrence. Sfold web server for statistical folding and rational design of nucleic acids. *Nucleic Acids Research*, 32:W135–W141, 2004.

[DCL05] Ye Ding, Chi Yu Chan, and Charles E. Lawrence. RNA secondary structure prediction by centroids in a Boltzmann weighted ensemble. *RNA*, 11:1157–1166, 2005.

[DE04] Robin D. Dowell and Sean R. Eddy. Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction. *BMC Bioinformatics*, 5:71, 2004.

[Din06] Ye Ding. Statistical and bayesian approaches to RNA secondary structure prediction. *RNA*, 12:323–331, 2006.

[DL03] Ye Ding and Charles E. Lawrence. A statistical sampling algorithm for RNA secondary structure prediction. *Nucleic Acids Research*, 31(24):7280–7301, 2003.

[DP03] R. M. Dirks and N. A. Pierce. A partition function algorithm for nucleic acid secondary structure including pseudoknots. *J. Comput. Chem.*, 24:1664–1677, 2003.

[DP04] R. M. Dirks and N. A. Pierce. An algorithm for computing nucleic acid base-pairing probabilities including pseudoknots. *J. Comput. Chem.*, 25:1295–1304, 2004.

[DWB06] Chuong B. Do, Daniel A. Woods, and Serafim Batzoglou. CONTRAfold: RNA secondary structure prediction without physics-based models. *Bioinformatics*, 22(14):e90–e98, 2006.

[FH72] K. S. Fu and T. Huang. Stochastic grammars and languages. *International Journal of Computer and Information Sciences*, 1(2):135–170, 1972.

[GJBM+03] S. Griffiths-Jones, A. Bateman, M. Marshall, A. Khanna, and S. R. Eddy. Rfam: an RNA family database. *Nucleic Acids Res.*, 31(1):439–441, 2003.

[GJMM+05] S. Griffiths-Jones, S. Moxon, M. Marshall, A. Khanna, S.R. Eddy, and A. Bateman. Rfam: annotating non-coding RNAs in complete genomes. *Nucleic Acids Res.*, 33:D121–D124, 2005.

[Goo98] Joshua T. Goodman. *Parsing Inside-Out.* PhD thesis, Harvard University, Cambridge, Massachusetts, May 1998.

[Goo99] Joshua Goodman. Semiring parsing. *Computational Linguistics*, 25(4):573–605, 1999.

[GVR04] Robert Giegerich, Björn Voß, and Marc Rehmsmeier. Abstract shapes of RNA. *Nucleic Acids Research*, 32(16):4843–4851, 2004.

[GzS11] Robert Giegerich and Christian Hner zu Siederdissen. Semantics and ambiguity of stochastic rna family models. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8:499–516, 2011.

[HF71] T. Huang and K. S. Fu. On stochastic context-free languages. *Information Sciences*, 3:201–224, 1971.

[HFS$^+$94] I. L. Hofacker, W. Fontana, P. F. Stadler, L. S. Bonhoeffer, and P. Schuster. Fast folding and comparison of RNA secondary structures (the Vienna RNA package). *Monatsh Chem.*, 125:167–188, 1994.

[HKS$^+$09] Michiaki Hamada, Hisanori Kiryu, Kengo Sato, Toutai Mituyama, and Kiyoshi Asai. Prediction of RNA secondary structure using generalized centroid estimators. *Bioinformatics*, 25(4):465–473, 2009.

[Hof03] Ivo L. Hofacker. The Vienna RNA secondary structure server. *Nucleic Acids Research*, 31(13):3429–3431, 2003.

[JRG08] Stefan Janssen, Jens Reeder, and Robert Giegerich. Shape based indexing for faster search of RNA family databases. *BMC Bioinformatics*, 9(131), 2008.

[KH99] B. Knudsen and J. Hein. RNA secondary structure prediction using stochastic context-free grammars and evolutionary history. *Bioinformatics*, 15(6):446–454, 1999.

[KH03] B. Knudsen and J. Hein. Pfold: RNA secondary structure prediction using stochastic context-free grammars. *Nucleic Acids Research*, 31(13):3423–3428, 2003.

[McC90] J. S. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29:1105–1119, 1990.

[MSZT99] D. H. Mathews, J. Sabina, M. Zuker, and D. H. Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J. Mol. Biol.*, 288:911–940, 1999.

[NJ80] R. Nussinov and A. B. Jacobson. Fast algorithms for predicting the secondary structure of single-stranded RNA. *Proceedings of the National Academy of Science of the USA*, 77(11):6309–6313, 1980.

[NPGK78] R. Nussinov, G. Pieczenik, J. R. Griggs, and D. J. Kleitman. Algorithms for loop matchings. *SIAM Journal on Applied Mathematics*, 35:68–82, 1978.

[NS] Markus E. Nebel and Anika Scheid. Random generation of RNA secondary structures according to native distributions. Submitted.

[NS09] Markus E. Nebel and Anika Scheid. On quantitative effects of RNA shape abstraction. *Theory in Biosciences*, 128(4):211, 2009.

[RCM99] J. Rozenski, P.F. Crain, and J.A. McCloskey. The RNA modification database. *Nucleic Acids Research*, 27:196–197, 1999.

[RD94] Sean R.Eddy and Richard Durbin. RNA sequence analysis using covariance models. *Nucleic Acids Research*, 2(11):2079–2088, 1994.

[RE99] E. Rivas and S. R. Eddy. A dynamic programming algorithm for RNA structure prediction including pseudoknots. *J. Mol. Biol.*, 285:2053–2068, 1999.

[RE00] E. Rivas and S. R. Eddy. Secondary structure alone is generally not statistically significant for the detection of noncoding RNAs. *Bioinformatics*, 6:583–605, 2000.

[RG04] J. Reeder and R. Giegerich. Design, implementation and evaluation of a practical pseudoknot folding algorithm based on thermodynamics. *BMC Bioinformatics*, 5:104, 2004.

[RLE11] Elena Rivas, Raymond Lang, and Sean R. Eddy. A range of complex probabilistic models for rna secondary structure prediction that include the nearest neighbor model and more. *submitted*, 2011.

[RSZ04] J. Ruan, G.D. Stormo, and W. Zhang. An iterated loop matching approach to the prediction of RNA secondary structures with pseudoknots. *Bioinformatics*, 20(1):58–66, 2004.

[SBEB02] Maciej Szymanski, Miroslawa Z. Barciszewska, Volker A. Erdmann, and Jan Barciszewski. 5s ribosomal RNA database. *Nucleic Acids Res.*, 30:176–178, 2002.

[SHB$^+$98] M. Sprinzl, C. Horn, M. Brown, A. Ioudovitch, and S. Steinberg. Compilation of tRNA sequences and sequences of tRNA genes. *Nucleic Acids Res.*, 26:148–153, 1998.

[SVR$^+$06] Peter Steffen, Björn Voß, Marc Rehmsmeier, Jens Reeder, and Robert Giegerich. RNAshapes 2.1.1 manual, February 2006.

[VC85] G. Viennot and M. Vauchaussade De Chaumont. Enumeration of RNA secondary structures by complexity. *Mathematics in medicine and biology, Lecture Notes in Biomathematics*, 57:360–365, 1985.

[Wat78] M. S. Waterman. Secondary structure of single-stranded nucleic acids. *Advances in Mathematics Supplementary Studies*, 1:167–212, 1978.

[WFHS99] S. Wuchty, W. Fontana, I. Hofacker, and P. Schuster. Complete suboptimal folding of RNA and the stability of secondary structures. *Biopolymers*, 49:145–165, 1999.

[WN10] Frank Weinberg and Markus E. Nebel. Extending stochastic context-free grammars for an application in bioinformatics. In *4th International Conference on Language and Automata Theory and Applications (LATA2010)*, 2010.

[XSB$^+$98] T. Xia, J. SantaLucia Jr., M. E. Burkard, R. Kierzek, S. J. Schroeder, X. Jiao, C. Cox, and D. H. Turner. Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with Watson-Crick base pairs. *Biochemistry*, 37:14719–14735, 1998.

[ZMT99] M. Zuker, D. H. Mathews, and D. H. Turner. Algorithms and thermodynamics for RNA secondary structure prediction: A practical guide. In J. Barciszewski and B. F. C. Clark, editors, *RNA Biochemistry and Biotechnology*, NATO ASI Series, pages 11–43. Kluwer Academic Publishers, Dordrecht, NL, 1999.

[ZS81] M. Zuker and P. Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Research*, 9:133–148, 1981.

[Zuk89] M. Zuker. On finding all suboptimal foldings of an RNA molecule. *Science*, 244:48–52, 1989.

[Zuk03] M. Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.*, 31(13):3406–3415, 2003.

Supplementary Material

Sm-I Computing Inside and Outside Probabilities

In order the determine all inside and outside variables for a given sequence $r \in \mathcal{L}_r$, we decided to use the SCFG $\mathcal{G}_r$ as the basis for a special version of Earley's algorithm. In particular, we chose to rely on the formalism presented in [Goo98, Goo99] for describing parsers, which is called *semiring parsing*. The advantage of using an Earley-style parser description is that the corresponding semiring parser can handle general grammars, which means we do not have to transform the grammar $\mathcal{G}_r$ into Chomsky normal form (CNF). This is especially useful, since the number of productions of the CNF of grammar $\mathcal{G}_r$ would be huge. For this reason, computing the needed inside and outside values by the usual inside outside algorithm for grammars in CNF would be less efficient.

Sm-I.1 Notations

In the sequel, we number the nucleotides in a given RNA sequence r of length n in the usual $5' \rightarrow 3'$ direction (i.e., in the usual reading order from left to right), such that the corresponding RNA sequence can be written as $r_1 \dots r_n$. Equivalently, for a secondary structure s (in dot-bracket representation) of size n, we can write $s_1 \dots s_n$.

Moreover, for A an intermediate (or non-terminal) symbol of the considered grammar $\mathcal{G}_r$, let $\alpha_A(i,j)$ denote the inside variables (computed by the usual inside algorithm) and $\beta_A(i,j)$ denote the outside variables (computed by the usual outside algorithm) for a given word $r \in \mathcal{L}_r$ of size n, $1 \leq i,j \leq n$. Consequently, $\alpha_A(i,j)$ is the probability of a leftmost derivation that generates the subword $r_i \dots r_j$ (of a word $r \in \mathcal{L}_r = \mathcal{L}(\mathcal{G}_r)$) from the intermediate symbol A and $\beta_A(i,j)$ is the probability of a derivation[16] which, starting with the intermediate symbol S (the axiom of grammar $\mathcal{G}_r$), generates the sentential form $r_1 \dots r_{i-1}\ A\ r_{j+1} \dots r_n$.

Furthermore, we need to define a new set of productions that has to be used by our semiring parser in order to compute the desired inside and outside probabilities. This production set contains the so-called *dotted* rules that are considered by Earley's algorithm. It can easily be obtained by modifying the rule set $\mathcal{R}_{\mathcal{G}_r}$ of the grammar $\mathcal{G}_r$ in the following way: Introduce a new symbol $\bullet \notin \Sigma_{\mathcal{G}_r} \cup \mathcal{I}_{\mathcal{G}_r}$ that is used to mark the current position up to which the parsing has proceeded; according to the fact that Earley's algorithm parses input words from left to right, this symbols must thus be "shifted" from the leftmost position to the rightmost one in each production rule of the grammar used for parsing. For this reason, we replace each production $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r}$ of the form $\mathit{rule} = A \rightarrow \alpha_1 \dots \alpha_k$ with $\alpha_i \in \mathcal{I}_{\mathcal{G}_r} \cup \Sigma_{\mathcal{G}_r}$, $1 \leq i \leq k$, by $k+1$ new productions $\mathit{rule}_0 = A \rightarrow \bullet\alpha_1 \dots \alpha_k$, $\mathit{rule}_1 = A \rightarrow \alpha_1 \bullet \dots \alpha_k$, ..., $\mathit{rule}_{k-1} = A \rightarrow \alpha_1 \dots \bullet\, \alpha_k$ and $\mathit{rule}_k = A \rightarrow \alpha_1 \dots \alpha_k\bullet$; if $\mathit{rule} = A \rightarrow \epsilon$, it is replaced by the new production $\mathit{rule}_0 = A \rightarrow \epsilon\bullet$. The resulting dotted production set will be denoted by $\mathcal{R}_{\mathcal{G}_r,\bullet}$ in the sequel. Moreover, each set of $k+1$ productions that were derived from an original production $\mathit{rule} = A \rightarrow \alpha_1 \dots \alpha_k \in \mathcal{R}_{\mathcal{G}_r}$ will be denoted by $\mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule})$, such that $\bigcup_{\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r}} \mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule}) = \mathcal{R}_{\mathcal{G}_r,\bullet}$. Obviously, $\mathcal{R}_{\mathcal{G}_r,\bullet}$ contains exactly the rules that have to be considered by Earley's algorithm for grammar $\mathcal{G}_r$.

Last but not least, note that for defining the desired Earley-based semiring parser, we use an *item-based* parser description. Therefore, in contrast to the usual inside outside algorithm for the computing the inside values $\alpha_A(i,j)$ and outside values $\beta_A(i,j)$, $1 \leq i,j \leq n$, for A an intermediate symbol of the considered grammar and n the length of the input word, the corresponding semiring parser used in this work computes inside and outside values for so-called *items*. Here, items are defined by three components, having the form $[i, \mathrm{ind}(\mathit{rule}), j]$, where for a given input word $r \in \mathcal{L}_r$ of length n, i and j, $1 \leq i,j \leq n+1$, define positions in r (i.e., in front of the first character, in between two characters or after the last character). Additionally, $\mathrm{ind}(\mathit{rule})$ denotes the index of production $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r,\bullet}$ in an appropriate ordering (details will follow later) of production set $\mathcal{R}_{\mathcal{G}_r,\bullet}$. In fact, an item of the form $[i, \mathrm{ind}(A \rightarrow \alpha \bullet \beta), j]$ asserts that $A \Rightarrow \alpha\beta \overset{*}{\Rightarrow} r_i \dots r_{j-1}\beta$. Consequently, by semiring parsing, the inside and outside values are computed for each production $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r,\bullet}$ and not as needed for each non-terminal symbol $A \in \mathcal{I}_{\mathcal{G}_r}$. However, the needed inside and outside values $\alpha_A(i,j)$ and $\beta_A(i,j)$ can easily be derived from the corresponding inside and outside results for items $[i, \mathrm{ind}(A \rightarrow \gamma\bullet), j]$, as we will see later.

[16] Note that for the computation of this probability, one always summarizes over all corresponding derivation trees.

Sm-I.2 Deriving the Inside and Outside Values of Items

First, we want to describe how to compute the inside and outside values of all items by semiring parsing, using a corresponding item-based description of an Earley-style parser.

Sm-I.2.1 Inside Computation

To obtain the inside values of all items $[i, \mathrm{ind}(\mathit{rule}), j]$, $1 \leq i, j \leq n+1$ (for an RNA sequence r of size n) and $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r, \bullet}$, by semiring parsing based on Earley's algorithm, we can use the following formulae, which we derived according to [Goo98, Goo99]:

- Scanning:

  $$\mathrm{IN}[i, \mathrm{ind}(A \to \alpha w_j \bullet \beta), j+1] = \delta_{w_j, r_j} \cdot \mathrm{IN}[i, \mathrm{ind}(A \to \alpha \bullet w_j \beta), j]$$

  where for w_j an arbitrary terminal symbol of the underlying grammar $\mathcal{G}_r$ and r_j the (terminal) symbol read at position j of the input string r,

  $$\delta_{w_j, r_j} = \begin{cases} 1, & \text{if } w_j = r_j, \\ 0, & \text{if } w_j \neq r_j, \end{cases}$$

  according to the definition of Kronecker's delta.

- Prediction:

  $$\mathrm{IN}[j, \mathrm{ind}(B \to \bullet\gamma), j] = \begin{cases} \Pr(B \to \gamma), & \text{if } S \overset{*}{\Rightarrow} r_1 \dots r_{j-1} B \delta \text{ for some } \delta, \\ 0, & \text{else,} \end{cases}$$

  where $\Pr(\mathit{rule})$ denotes the probability of production $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r}$ as given by the SCFG $\mathcal{G}_r$.
  Note that this top down filtering is usually made by Earley's algorithm to ensure that only such items can be predicted that might later be used by the completion rule. However, this is not needed here, since for any superfluously predicted item, the resulting probability will later be set to 0 by a scan. Thus, we can simply predict all items by

  $$\mathrm{IN}[j, \mathrm{ind}(B \to \bullet\gamma), j] = \Pr(B \to \gamma).$$

- Completion:

  $$\mathrm{IN}[i, \mathrm{ind}(A \to \alpha B \bullet \beta), j] = \sum_{i \leq k \leq j} \mathrm{IN}[i, \mathrm{ind}(A \to \alpha \bullet B\beta), k] \cdot \sum_{\mathit{rule}_B \in \mathcal{R}_B} \mathrm{IN}[k, \mathrm{ind}(\mathit{rule}_B), j],$$

  where $\mathcal{R}_B = \{\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r, \bullet} \mid \mathit{rule} = B \to \gamma\bullet\}$.

Moreover, the desired semiring parser algorithm for the correct computation of all inside values additionally requires the definition of a convenient ordering of the considered items $[i, \mathrm{ind}(\mathit{rule}), j]$, for $1 \leq i, j \leq n+1$ and $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r, \bullet}$, such that no item precedes any other item on which it depends. Details on how we derived the corresponding ordering used in this work will follow. In principle, we can define an ordering by first and last parameters $i, j \in \{1, \dots, n+1\}$ that matches the order of consideration of items induced by Earley's algorithm and especially an appropriate ordering of the considered rule set $\mathcal{R}_{\mathcal{G}_r, \bullet}$ by indices (p, q), for $p \in \{1, \dots, \mathrm{card}(\mathcal{R}_{\mathcal{G}_r})\}$ and $q \in \{0, \dots, k(p)\}$, where $k(p)$ denotes the conclusion length of the production $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r}$ indexed by p.
Based on the previously introduced formulae and the appropriate ordering that will be formally defined hereafter, we finally obtain Algorithm 1 that shows how to perform the complete inside computation.

Sm-I.2.2 Ordering of Items

According to [Goo98, Goo99], we initially need to define an ordering on the items $[i, \mathrm{ind}(\mathit{rule}), j]$, $1 \leq i, j \leq n+1$ for n the length of the input word and $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r, \bullet}$. In fact, we have to take care that when successively computing the values of all items, no item precedes any item on which it depends. For this reason, in [Goo98, Goo99], each item x is associated with a "bucket" B; they write $\mathit{bucket}(x) = B$. The buckets have to be ordered as follows: If item y depends on item x, then $\mathit{buckets}(x) \leq \mathit{buckets}(y)$. There are two types of buckets: looping buckets and non-looping buckets. In fact, if items x and y depend

Algorithm 1 Computation of Inside Values

```
Require: RNA sequence $r \in \mathcal{L}_r$ of length $n \geq 1$,
         set $\mathcal{R}_{\mathcal{G}_r,\bullet}$ of production rules used by Earley's algorithm for parsing $r$ with $\mathcal{G}_r$, and
         probabilities $\Pr(\mathit{rule})$ of the productions $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r}$, trained on RNA structure data.
  for $j = 1 \ldots n+1$ do
    for $i = j \ldots 1$ do
      for $p = 1 \ldots \mathrm{card}(\mathcal{R}_{\mathcal{G}_r})$ do
        for $q = 0 \ldots k(p)$ do
          $\mathit{rule} = \mathrm{ind}^{-1}(p,q)$ /*$\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r,\bullet}$ is the rule having index $(p,q)$ in our ordering.*/
          if $\mathit{rule} = A \to \alpha w_{j-1} \bullet \beta$ then
            /* Scanning: */
            $\mathrm{IN}[i,(p,q),j] = \delta_{w_{j-1},r_{j-1}} \cdot \mathrm{IN}[i,(p,q-1),j-1]$
          else if $\mathit{rule} = B \to \bullet\gamma$ then
            /* Prediction: */
            $\mathrm{IN}[j,(p,q),j] = \Pr(B \to \gamma)$
          else if $\mathit{rule} = A \to \alpha B \bullet \beta$ then
            /* Completion: */
            $\mathrm{IN}[i,(p,q),j] = \sum_{i \leq k \leq j} \left(\mathrm{IN}[i,(p,q-1),k] \cdot \left(\sum_{\mathit{rule}_B \in \mathcal{R}_B} \mathrm{IN}[k,\mathrm{ind}(\mathit{rule}_B),j]\right)\right)$
          end if
        end for
      end for
    end for
  end for
```

(directly or indirectly) on each other, then they are both associated with a special looping bucket B, such that $\mathit{bucket}(x) = B = \mathit{bucket}(y)$. A bucket is also called looping bucket if an item in it depends on itself. Otherwise, the bucket is called non-looping. If item x is associated with a non-looping bucket, then its value can easily be computed, as this value depends only on the values of items in earlier buckets. However, in the case of item x being associated with a looping bucket, the computation is much more complex, which is due to the fact that the value of x then depends potentially on the values of other items in the same bucket. In fact, this means that infinite loops may occur, for two different reasons: First, if the values of two items in the same bucket are mutually dependent, or second if an item depends on its own value. Although such infinite loops may require computation of infinite sums, there exists a way to efficiently compute or approximate them, as shown in [Goo98, Goo99].

Fortunately, as the SCFG $\mathcal{G}_r$ considered in this work is loop-free, each item $[i, \mathrm{ind}(\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r,\bullet}), j]$ can be associated with a non-looping bucket B (of size one). Thus, considering the restriction that no item precedes any item on which it depends, an ordering on the items $[i, \mathrm{ind}(\mathit{rule}), j]$ can be defined by appropriately iterating over positions i and j, respectively, as well as by using a suitable ordering (indexing) of the elements in $\mathcal{R}_{\mathcal{G}_r,\bullet}$. Since we use an Earley-style parser, it is obvious that in order to calculate all values of items $[i, \mathrm{ind}(\mathit{rule}), j]$, $1 \leq i, j \leq n+1$ and $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r,\bullet}$, we first have to iterate over all values j from 1 to $n+1$. This means we "shift" the symbol $\bullet$[17] from left to right. For each value of $j \in \{1, \ldots, n+1\}$, we then have to iterate over all values i from j down to 1. Thus, we can first make a prediction for $i = j$ and then scanning or completion steps for $i < j$. However, the problem of finding an appropriate ordering of $\mathcal{R}_{\mathcal{G}_r,\bullet}$ that has to be applied for every pair of fixed positions i and j in order to derive the values for items $[i, \mathrm{ind}(\mathit{rule}), j]$, $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r,\bullet}$, is more complicated.

In this work, the ordering of the rules in $\mathcal{R}_{\mathcal{G}_r,\bullet}$ is defined by index values (p, q), given as follows:

- The first index $p \in \{1, \ldots, \mathrm{card}(\mathcal{R}_{\mathcal{G}_r})\}$ corresponds to a set of productions $\mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule}) \subset \mathcal{R}_{\mathcal{G}_r,\bullet}$ (the one that was derived from production $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r}$) and
- the corresponding second index $q \in \{0, \ldots, \mathrm{card}(\mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule}))\}$ corresponds to a single production $\mathit{rule}_q \in \mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule})$ (the one in which symbol $\bullet$ occurs after the qth symbol in the conclusion, see above).

Obviously, this ordering within the sets $\mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule})$ is appropriate, since if $\mathit{rule} = A \to \alpha B \beta$ is indexed by $p \in \{1, \ldots, \mathrm{card}(\mathcal{R}_{\mathcal{G}_r})\}$, then item $[i, (p,q) = \mathrm{ind}(A \to \alpha B \bullet \beta), j]$ depends on item $[i, (p, q-1) = \mathrm{ind}(A \to \alpha \bullet B \beta), j']$ for $j' \leq j$. Consequently, it remains to find a suitable distinct index $p \in \{1, \ldots, \mathrm{card}(\mathcal{R}_{\mathcal{G}_r})\}$ for

[17] Recall that symbol $\bullet$ is used to mark the current position j, $1 \leq j \leq n+1$, in the input word up to which the parsing has proceeded.

any set $\mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule})$ corresponding to the original production $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r}$, such that the resulting ordering ensures that no item precedes any item on which it depends.

It is easy to see that for predictions and scanning steps, no problems can occur due to our ordering (implied by index q) within any set $\mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule})$. Thus, the center of attention has to be laid on the completion steps. In fact, suppose the value of an item $[i,(p,q) = \mathrm{ind}(A \to \alpha B \bullet \beta), j]$ has to be computed by completion. Then, this value depends on the values of items $[i,(p,q-1) = \mathrm{ind}(A \to \alpha \bullet B\beta), k]$ and $[k, \mathrm{ind}(\mathit{rule}_B \in \mathcal{R}_B), j]$, for $i \leq k \leq j$. Whereas in all cases, the value of $[i,(p,q-1),k]$ has been computed at this point (due to our ordering of $\mathcal{R}_{\mathcal{G}_r,\bullet}(A \to \alpha B \beta)$ and since $k \leq j$), problems may arise for $[k, \mathrm{ind}(\mathit{rule}_B \in \mathcal{R}_B), j]$. Particularly, if α can not be the empty word, i.e., if $|\alpha| \geq 1$ holds, then we only have to consider $i+1 \leq k \leq j$, in which cases values of items $[k, \mathrm{ind}(\mathit{rule}_B \in \mathcal{R}_B), j]$ have already been determined in previous iterations, since $k > i$. However, if α can be empty, then $[i,(p,q) = \mathrm{ind}(A \to \alpha B \bullet \beta), j]$ also depends on $[i,(p',q') = \mathrm{ind}(\mathit{rule}_B = B \to \gamma \bullet \in \mathcal{R}_B), j]$. Thus, $B \to \gamma$ has to be considered before $A \to \alpha B \beta$, which implies $p' < p$ must hold. In fact, as this holds for any $\mathit{rule}_B \in \mathcal{R}_B$, we can conclude that if α can be empty, then in an appropriate ordering, $A \to \alpha B \beta \in \mathcal{R}_{\mathcal{G}_r}$ has to be placed after all productions $B \to \gamma \in \mathcal{R}_{\mathcal{G}_r}$ that have premise B.

According to these observations, the desired ordering can easily be constructed in the following way: Start by assigning the smallest indices $p \in \{1, \ldots, \mathrm{card}(\mathcal{R}_{\mathcal{G}_r})\}$ to productions of the form $\mathit{rule} = I \to t\delta$, where the first symbol t of the conclusion is any terminal symbol from $\Sigma_{\mathcal{G}_r}$. Then, assign the remaining indices to the other sets $\mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule})$, for $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r}$, taking into account the previously discussed restrictions. For the sake of simplicity, let us first consider the grammar $\mathcal{G}_s$ that models the language $\mathcal{L}_s$ of all secondary structures. For this grammar, we could for example use the following ordering of the corresponding rule set $\mathcal{R}_{\mathcal{G}_s}$, i.e., the following ordering by first indices $p \in \{1, \ldots, \mathrm{card}(\mathcal{R}_{\mathcal{G}_s})\}$:

Index p	Rule r	Index p	Rule r	Index p	Rule r	Index p	Rule r
1	$Z \to \circ$,	2	$A \to (^{m_s}L)^{m_s}$,	3	$P \to (L)$,		
4	$C \to ZC$,	5	$C \to Z$,	6	$H \to ZH$,	7	$H \to Z$,
8	$B \to ZB$,	9	$B \to Z$,	10	$U \to ZU$,	11	$U \to \epsilon$,
12	$T \to C$,	13	$T \to A$,	14	$T \to CA$,		
15	$T \to AT$,	16	$T \to CAT$,	17	$F \to Z^{m_h-1}H$,		
18	$G \to BA$,	19	$G \to AB$,	20	$G \to BAB$,		
21	$M \to UAO$,	22	$O \to UAN$,	23	$N \to UAN$,	24	$N \to U$,
25	$L \to F$,	26	$L \to P$,	27	$L \to G$,	28	$L \to M$,
29	$S \to T$.						

The derivation of a corresponding ordering for the considered SCFG $\mathcal{G}_r$ generating all RNA sequences is straightforward. Thus, we have defined an appropriate ordering of $\mathcal{R}_{\mathcal{G}_r,\bullet}$ by indices (p,q), for $p \in \{1, \ldots, \mathrm{card}(\mathcal{R}_{\mathcal{G}_r})\}$ and $q \in \{0, \ldots, k(p)\}$, where $k(p) = \mathrm{card}(\mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule}))$ if $\mathcal{R}_{\mathcal{G}_r,\bullet}(\mathit{rule})$ can be found under index p.

Sm-I.2.3 Outside Computation

Once the inside values have been computed (with Algorithm 1), the corresponding outside values of all items $[i, \mathrm{ind}(\mathit{rule}), j]$, $1 \leq i,j \leq n+1$ (for an RNA sequence r of size n) and $\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r,\bullet}$ can be calculated with Algorithm 2. This Earley-based semiring parser algorithm uses the reversed previously introduced ordering of items and makes use of the following formulae for the outside computations (for details, we refer to [Goo98, Goo99]):

- Scanning (reverse):

$$\mathrm{OUT}[i, \mathrm{ind}(A \to \alpha \bullet w_j \beta), j] = \delta_{w_j, r_j} \cdot \mathrm{OUT}[i, \mathrm{ind}(A \to \alpha w_j \bullet \beta), j+1].$$

- Prediction (reverse):
  There is nothing to do, since this value is obtained while performing a (reverse) completion computation.

- Completion (reverse):

$$\begin{aligned}\mathrm{OUT}[i, \mathrm{ind}(A \to \alpha \bullet B\beta), k] = \mathrm{OUT}[i, \mathrm{ind}(A \to \alpha \bullet B\beta), k] + \\ \mathrm{OUT}[i, \mathrm{ind}(A \to \alpha B \bullet \beta), j] \cdot \sum_{\mathit{rule}_B \in \mathcal{R}_B} \mathrm{IN}[k, \mathrm{ind}(\mathit{rule}_B), j]\end{aligned}$$

Algorithm 2 Computation of Outside Values

```
Require: RNA sequence $r \in \mathcal{L}_r$ of length $n \geq 1$,
         set $\mathcal{R}_{\mathcal{G}_r,\bullet}$ of production rules used by Earley's algorithm for parsing $r$ with $\mathcal{G}_r$, and
         the corresponding inside values (computed by Algorithm 1).
  $\mathrm{OUT}[1, \mathrm{ind}(S \to T\bullet), n+1] = 1$
  for $j = n+1 \ldots 1$ do
    for $i = 1 \ldots j$ do
      for $p = \mathrm{card}(\mathcal{R}_{\mathcal{G}_r}) \ldots 1$ do
        for $q = k(p) \ldots 0$ do
          $\mathit{rule} = \mathrm{ind}^{-1}(p,q)$ /*$\mathit{rule} \in \mathcal{R}_{\mathcal{G}_r,\bullet}$ is the rule having index $(p,q)$ in our ordering.*/
          if $\mathit{rule} = A \to \alpha w_j \bullet \beta$ then
            /* Scanning (reverse): */
            $\mathrm{OUT}[i,(p,q-1),j] = \delta_{w_j,r_j} \cdot \mathrm{OUT}[i,(p,q),j+1]$
          else if $\mathit{rule} = B \to \bullet\gamma$ then
            /* Prediction (reverse): */
            do nothing
          else if $\mathit{rule} = A \to \alpha B \bullet \beta$ then
            /* Completion (reverse): */
            for $k = i \ldots j$ do
              $\mathrm{OUT}[i,(p,q-1),k] =$
                  $\mathrm{OUT}[i,(p,q-1),k] + \mathrm{OUT}[i,(p,q),j] \cdot \left(\sum_{\mathit{rule}_B \in \mathcal{R}_B} \mathrm{IN}[k, \mathrm{ind}(\mathit{rule}_B), j]\right)$
              for $\mathit{rule}_B \in \mathcal{R}_B$ do
                $\mathrm{OUT}[k, \mathrm{ind}(\mathit{rule}_B), j] =$
                    $\mathrm{OUT}[k, \mathrm{ind}(\mathit{rule}_B), j] + \mathrm{OUT}[i,(p,q),j] \cdot \mathrm{IN}[i,(p,q-1),k]$
              end for
            end for
          end if
        end for
      end for
    end for
  end for
```

and

$$\mathrm{OUT}[k, \mathrm{ind}(\mathit{rule}_B), j] = \mathrm{OUT}[k, \mathrm{ind}(\mathit{rule}_B), j] + \mathrm{OUT}[i, \mathrm{ind}(A \to \alpha B \bullet \beta), j] \cdot \mathrm{IN}[i, \mathrm{ind}(A \to \alpha \bullet B\beta), k],$$

for $i \leq k \leq j$ and $\mathit{rule}_B \in \mathcal{R}_B$.

Since the number of production rules considered for the inside and outside computations is given by $\mathrm{card}(\mathcal{R}_{\mathcal{G}_r,\bullet})$ and is thus not dependent on the input size, Algorithms 1 and 2 need cubic time and quadratic space in the worst-case.

Sm-I.3 Deriving the Needed Inside and Outside Probabilities

Finally, since for a given sequence $r \in \mathcal{L}_r$ of length n, an item of the form $[i, \mathrm{ind}(A \to \alpha\bullet), j+1]$, $1 \leq i,j \leq n+1$, asserts that $A \Rightarrow \alpha \overset{*}{\Rightarrow} r_i \ldots r_j$, it is easy to see that

$$\sum_{\mathit{rule}=A\to\alpha\bullet} \mathrm{IN}[i, \mathrm{ind}(\mathit{rule}), j+1] = \sum_{\mathit{rule}\in\mathcal{R}_A} \mathrm{IN}[i, \mathrm{ind}(\mathit{rule}), j+1] = \alpha_A(i,j)$$

is the probability of a leftmost derivation that generates the subword $r_i \ldots r_j$ of r from the intermediate symbol A. Furthermore, recall that $\beta_A(i,j)$ is defined as the probability of a derivation which, starting with the intermediate symbol S (the axiom of the grammar $\mathcal{G}_r$), generates the expression $r_1 \ldots r_{i-1}\ A\ r_{j+1} \ldots r_n$. For this outside probability, it obviously does not matter what subword $r_i \ldots r_j$ of r is derived from intermediate symbol A, i.e., it is independent on which rule $A \to \alpha\bullet \in \mathcal{R}_A$ generates subword $r_i \ldots r_j$. Consequently, for $\mathit{rule} = A \to \alpha\bullet \in \mathcal{R}_A$, the outside value for item $[i, \mathrm{ind}(\mathit{rule}), j+1]$ is either equal to zero (if $r_i \ldots r_j$ can not be derived from non-terminal A using production *rule*), or it is equal to the outside value for any items $[i, \mathrm{ind}(\mathit{rule}'), j+1]$, where $\mathit{rule}' \in \mathcal{R}_A$ and $\mathrm{OUT}[i, \mathrm{ind}(\mathit{rule}'), j+1] \neq 0$

(which means that production $rule'$ can be used to generate subword $r_i \dots r_j$ of r from A). Accordingly, the needed outside probability for symbol A is equal to one of the non-zero values (if any) of the corresponding production rules with premise A, which can be written as:

$$\max_{rule=A\to\alpha\bullet} \mathrm{OUT}[i, \mathrm{ind}(rule), j+1] = \max_{rule\in\mathcal{R}_A} \mathrm{OUT}[i, \mathrm{ind}(rule), j+1] = \beta_A(i,j).$$

Thus, for any given RNA sequence $r \in \mathcal{L}_r$ of size n, we can derive the desired inside and outside probabilities $\alpha_A(i,j)$ and $\beta_A(i,j)$, for each $A \in \mathcal{I}_{\mathcal{G}_r}$ and $1 \leq i,j \leq n$, by computing the inside and outside values of all items by semiring parsing based on an Earley-style parser for the SCFG $\mathcal{G}_r$ and afterwards using the results for each $rule \in \mathcal{R}_{\mathcal{G}_r,\bullet}$ of the form $rule = A \to \alpha\bullet$ to obtain the corresponding probabilities for each $A \in \mathcal{I}_{\mathcal{G}_r}$ (in the previously described way). Consequently, for sequence r of size n, there result cubic time complexity and quadratic memory requirements for the computation of all probabilities $\alpha_A(i,j)$ and $\beta_A(i,j)$, $A \in \mathcal{I}_{\mathcal{G}_r}$ and $1 \leq i,j \leq n$.

Sm-II Details of the Sampling Algorithm

In this section, we first present equations for computing the needed sampling probabilities for all considered cases (except for exterior loops, since they have already been presented in Section 3.2.1). Afterwards, we give a detailed description of the corresponding sampling algorithm, including detailed information on how to use the respective sampling probabilities. Note that these parts are written in a similar way as the corresponding section in [DL03], in order to illustrate the similarities and differences that arise when computing the sampling probabilities according to either approach.

Sm-II.1 Equations for Computation of Sampling Probabilities

Basically, the definitions of the needed sampling probabilities for all regular loop types can be derived in the same way as those already presented in Section 3.2.1 for exterior loops – by using only the corresponding inside outside values and the probabilities of the production rules of the considered SCFG.

Sm-II.1.1 Sampling Probabilities for Substructures Between a Given Base Pair

Given a base pair $r_i.r_j$, then this pair can either be the closing base pair of a hairpin loop, the exterior pair of a base pair stack, the closing pair of a bulge or an interior loop, or close a multibranched loop. For all of these cases, the corresponding probabilities are given as follows:

$$\begin{aligned}
Q_{ij}^{HL}(i,j) &= \frac{1}{q_{ij}(i,j)} \cdot \beta_L(i+1,j-1) \cdot (\alpha_F(i+1,j-1) \cdot \Pr(L \to F)),\\
Q_{ij}^{SP}(i,j) &= \frac{1}{q_{ij}(i,j)} \cdot \beta_L(i+1,j-1) \cdot (\alpha_P(i+1,j-1) \cdot \Pr(L \to P)),\\
Q_{ij}^{BI}(i,j) &= \frac{1}{q_{ij}(i,j)} \cdot \beta_L(i+1,j-1) \cdot (\alpha_G(i+1,j-1) \cdot \Pr(L \to G)),\\
Q_{ij}^{ML}(i,j) &= \frac{1}{q_{ij}(i,j)} \cdot \beta_L(i+1,j-1) \cdot (\alpha_M(i+1,j-1) \cdot \Pr(L \to M)).
\end{aligned}$$

Here, we have to use the normalizing factor

$$q_{ij}(i,j) = \beta_L(i+1,j-1) \cdot \alpha_L(i+1,j-1).$$

Thus, $Q_{ij}^{HL}(i,j)$, $Q_{ij}^{SP}(i,j)$, $Q_{ij}^{BI}(i,j)$ and $Q_{ij}^{ML}(i,j)$ is the sampling probability for a hairpin loop, base pair stack, bulge or interior loop and multibranched loop, respectively, where for mutually exclusive and exhaustive cases, $Q_{ij}^{HL}(i,j) + Q_{ij}^{SP}(i,j) + Q_{ij}^{BI}(i,j) + Q_{ij}^{ML}(i,j) = 1$ holds.

Sm-II.1.2 Sampling Probabilities for Bulge and Interior Loops

For sampling bulge and interior loops corresponding to the PF approach, we would have to use the following probabilities:

$$P_{hl}^{BIL}(i,j,h,l) = \begin{cases} P_{hl}^{B1}(i,j,h), & \text{if } h > i+1 \text{ and } l = j-1,\\ P_{hl}^{B2}(i,j,l), & \text{if } h = i+1 \text{ and } l < j-1,\\ P_{hl}^{IL}(i,j,h), & \text{if } h > i+1 \text{ and } l < j-1,\\ 0, \text{ else}, \end{cases}$$

where

$$\begin{aligned}
P_{hl}^{B1}(i,j,h) &= \frac{1}{Q_{ij}^{BI}(i,j)\cdot q_{ij}(i,j)}\cdot\beta_L(i+1,j-1)\cdot\Pr(L\to G)\\
&\qquad\times\left(\alpha_B(i+1,h-1)\cdot\alpha_A(h,j-1)\cdot\Pr(G\to BA)\right),\\
P_{hl}^{B2}(i,j,l) &= \frac{1}{Q_{ij}^{BI}(i,j)\cdot q_{ij}(i,j)}\cdot\beta_L(i+1,j-1)\cdot\Pr(L\to G)\\
&\qquad\times\left(\alpha_A(i+1,l)\cdot\alpha_B(l+1,j-1)\cdot\Pr(G\to AB)\right),\\
P_{hl}^{IL}(i,j,h,l) &= \frac{1}{Q_{ij}^{BI}(i,j)\cdot q_{ij}(i,j)}\cdot\beta_L(i+1,j-1)\cdot\Pr(L\to G)\\
&\qquad\times\left(\alpha_B(i+1,h-1)\cdot\alpha_A(h,l)\cdot\alpha_B(l+1,j-1)\cdot\Pr(G\to BAB)\right).
\end{aligned}$$

After the case of bulge or interior loop was sampled, $\{P_{hl}^{BIL}(i,j,h,l)\}$ would then be used for sampling h and l (together in one single sampling step) and for mutually exclusive and exhaustive cases, $\sum_{h=(i+1)}^{j-\min_{\text{ps}}}\sum_{l=(h-1)+\min_{\text{ps}}}^{(j-1)} P_{hl}^{BIL}(i,j,h,l)=1$ (under the condition that $Q_{ij}^{BI}(i,j)>0$).
However, to ensure that the sampling algorithm runs in cubic time, we would then have to disregard long bulge and interior loops by using a constant $\max_{BL}$ – just like with PFs[18]. Nevertheless, we do *not* need to apply this restriction if we sample h and l one after the other with the following probabilities:

$$\begin{aligned}
P_{hj}^{BI}(i,j,h) &= \frac{1}{p^{BI}(i,j)}\cdot\beta_G(i+1,j-1)\cdot\left(\alpha_B(i+1,h-1)\cdot\alpha_A(h,j-1)\cdot\Pr(G\to BA)\right),\\
P_{il}^{BI}(i,j,l) &= \frac{1}{p^{BI}(i,j)}\cdot\beta_G(i+1,j-1)\cdot\left(\alpha_A(i+1,l)\cdot\alpha_B(l+1,j-1)\cdot\Pr(G\to AB)\right),\\
P_{hl}^{BI}(i,j,h) &= \frac{1}{p^{BI}(i,j)}\cdot\beta_G(i+1,j-1)\cdot\left(\alpha_B(i+1,h-1)\cdot\alpha_{AB}(h-1,j)\cdot\Pr(G\to BAB)\right),\\
\widehat{P}_{hl}^{BI}(j,h,l) &= \frac{1}{\alpha_{AB}(h-1,j)}\cdot\left(\alpha_A(h,l)\cdot\alpha_B(l+1,j-1)\right),
\end{aligned}$$

where

$$\alpha_{AB}(i,j)=\sum\nolimits_{l=i+\min_{\text{ps}}}^{(j-2)}\left(\alpha_A(i+1,l)\cdot\alpha_B(l+1,j-1)\right)$$

and

$$p^{BI}(i,j)=\beta_G(i+1,j-1)\cdot\alpha_G(i+1,j-1).$$

Obviously, $\{P_{hj}^{BI}(i,j,h)\}$ and $\{P_{il}^{BI}(i,j,l)\}$ are the sampling probabilities for bulges on the left and bulges on the right, respectively. Furthermore, $\{P_{hl}^{BI}(i,j,h)\}$ are the probabilities for first sampling h for interior loops and $\{\widehat{P}_{hl}^{BI}(j,h,l)\}$ are the probabilities for sampling l after h is sampled (for interior loops).
Since the probabilities of all mutually exclusive and exhaustive cases sum up to 1, we have $\sum_{h=(i+2)}^{j-\min_{\text{ps}}} P_{hj}^{BI}(i,j,h)+\sum_{l=i+\min_{\text{ps}}}^{(j-2)} P_{il}^{BI}(i,j,l)+\sum_{h=(i+2)}^{j-\min_{\text{ps}}-1} P_{hl}^{BI}(i,j,h)=1$, and, under the condition that $P_{hl}^{BI}(i,j,h)>0$, also $\sum_{l=(h-1)+\min_{\text{ps}}}^{(j-2)} \widehat{P}_{hl}^{BI}(j,h,l)=1$.

Sm-II.1.3 Sampling Probabilities for Multiloops

In the case of a multibranched loop, the probabilities for sampling the first accessible base pair $r_{h_1}.r_{l_1}$ within this loop can be obtained by considering the intermediate symbols of $\mathcal{G}_r$ that generate (parts of) multiloops. More specifically, we first sample h and l according to the following conditional probabilities:

$$\begin{aligned}
P_{hl}^{M_1}(i,j,h) &= \frac{1}{p^{M_1}(i,j)}\cdot\beta_M(i+1,j-1)\cdot\left(\alpha_U^{*}(i+1,h-1)\cdot\alpha_{AO}(h,j)\cdot\Pr(M\to UAO)\right),\\
\widehat{P}_{hl}^{M_1}(j,h,l) &= \frac{1}{\alpha_{AO}(h,j)}\cdot\left(\alpha_A(h,l)\cdot\alpha_O(l+1,j-1)\right),
\end{aligned}$$

where

$$\alpha_{AO}(h,j)=\sum\nolimits_{l=(h-1)+\min_{\text{ps}}}^{(j-1)-\min_{\text{ps}}}\left(\alpha_A(h,l)\cdot\alpha_O(l+1,j-1)\right)$$

and

$$p^{M_1}(i,j)=\beta_M(i+1,j-1)\cdot\alpha_M(i+1,j-1).$$

[18] Note that when using the PF approach based on thermodynamics, h and l have to be sampled at once, since the free energy of a bulge or interior loops strongly depends on both the closing pair $r_i.r_j$ and the accessible pair $r_h.r_l$.

Note that we have to take care of the ϵ-rule $U \to \epsilon$, which implies that symbol U may generate words of size zero. For this reason, $h = i+1$ could be chosen, implying $h-1 < i+1$. However, $\alpha_U(i+1, h-1)$ is only defined for $i+1 \leq h-1$. To fix this problem, we have used the term $\alpha_U^*(i+1, h-1)$ instead of $\alpha_U(i+1, h-1)$ in the previous two definitions, which is given as follows:

$$\alpha_U^*(i+1, h-1) = \begin{cases} \alpha_U(i+1, h-1), & \text{if } i+1 \leq h-1, \\ \Pr(U \to \epsilon), & \text{if } i+1 > h-1. \end{cases}$$

Thus, $\{\widehat{P}_{hl}^{M_1}(j,h,l)\}$ are probabilities for sampling l after $h \geq i+1$ is sampled with probabilities $\{P_{hl}^{M_1}(i,j,h)\}$. For mutually exclusive and exhaustive cases, we have $\sum_{h=(i+1)}^{j-2\cdot\min_{\text{ps}}} P_{hl}^{M_1}(i,j,h) = 1$, and accordingly, $\sum_{l=(h-1)+\min_{\text{ps}}}^{(j-1)-\min_{\text{ps}}} \widehat{P}_{hl}^{M_1}(j,h,l) = 1$. Sampling both h and l yields the first accessible base pair $r_{h_1}.r_{l_1} := r_h.r_l$ (which closes the first helix radiating out from this multiloop).

In order to sample the second accessible base pair $r_{h_2}.r_{l_2}$, we consider the remaining structure fragment $R_{(l_1+1)(j-1)}$ (between the $3'$ base r_{l_1} of the first accessible base pair $r_{h_1}.r_{l_1}$ and the $3'$ base r_j of the closing base pair $r_i.r_j$ of the considered multiloop). In fact, for any $k \geq 1$, the probabilities for sampling the $(k+1)$th accessible base pair $r_{h_{k+1}}.r_{l_{k+1}}$ within this multibranched loop are computed by considering the structure fragment $R_{(l_k+1)(j-1)}$ and using the corresponding inside and outside variables for some specific multiloop generating intermediate symbols of the grammar $\mathcal{G}_r$. More specifically, we first sample h and l according to conditional probabilities, which are defined as follows:

$$P_{hl}^{M_{k+1}}(l_k,j,h) = \frac{1}{p^{M_{k+1}}(l_k,j)} \cdot \beta_X(l_k+1, j-1) \cdot \left(\alpha_U^*(l_k+1, h-1) \cdot \alpha_{AN}(h,j) \cdot \Pr(X \to UAN)\right),$$
$$\widehat{P}_{hl}^{M_{k+1}}(j,h,l) = \frac{1}{\alpha_{AN}(h,j)} \cdot \left(\alpha_A(h,l) \cdot \alpha_N^*(l+1, j-1)\right),$$

where

$$\alpha_{AN}(h,j) = \sum\nolimits_{l=(h-1)+\min_{\text{ps}}}^{(j-1)} \left(\alpha_A(h,l) \cdot \alpha_N^*(l+1,j-1)\right)$$

and

$$p^{M_{k+1}}(l_k,j) = \begin{cases} \beta_O(l_k+1, j-1) \cdot \alpha_O(l_k+1, j-1), & \text{if } (k+1) = 2, \\ \beta_N(l_k+1, j-1) \cdot \alpha_N(l_k+1, j-1) - \\ \beta_N(l_k+1, j-1) \cdot \left(\alpha_U(l+1, j-1) \cdot \Pr(N \to U)\right), & \text{if } (k+1) \geq 3, \end{cases}$$

as well as

$$X = \begin{cases} O, & \text{if } (k+1) = 2, \\ N, & \text{if } (k+1) \geq 3. \end{cases}$$

Again, we have used α_U^* instead of α_U and α_N^* instead of α_N, which is defined as

$$\alpha_N^*(l+1, j-1) = \begin{cases} \alpha_N(l+1, j-1), & \text{if } l+1 \leq j-1, \\ \Pr(N \to U) \cdot \Pr(U \to \epsilon), & \text{if } l+1 > j-1, \end{cases}$$

in order to take care of possible cases where U and/or N generate words of size zero. According to these definitions, $\{\widehat{P}_{hl}^{M_{k+1}}(j,h,l)\}$ are probabilities for sampling l after $h \geq l_k+1$ is sampled with probabilities $\{P_{hl}^{M_{k+1}}(l_k,j,h)\}$ and again, for mutually exclusive and exhaustive cases, we have $\sum_{h=(l_k+1)}^{j-\min_{\text{ps}}} P_{hl}^{M_{k+1}}(l_k,j,h) = 1$, and $\sum_{l=(h-1)+\min_{\text{ps}}}^{(j-1)} \widehat{P}_{hl}^{M_{k+1}}(j,h,l) = 1$. By sampling both h and l, we obtain the desired $(k+1)$th accessible base pair $r_{h_{k+1}}.r_{l_{k+1}} := r_h.r_l$ (which closes the $(k+1)$th helix radiating out from this multiloop).

According to the definition of multibranched loops, we now have to address two different cases: either the considered multiloop contains no additional accessible base pair, or there is at least one more base pair accessible from the closing pair $r_i.r_j$. These two mutually exclusive cases are addressed by the following two probabilities: Conditional on the sampled values for h_k and l_k (for the kth accessible base pair $r_{h_k}.r_{l_k}$ in the considered multiloop), $k \geq 2$, we consider the following "decision" probability for no additional accessible base pairs on the structure fragment $R_{(l_k+1)(j-1)}$ (i.e., between the $3'$ base r_{l_k} of the kth accessible base pair $r_{h_k}.r_{l_k}$ and the $3'$ base r_j of the closing base pair $r_i.r_j$):

$$P_{01}^{M_{k+1}}(l_k,j) = \frac{1}{p_{01}(l_k,j)} \cdot \beta_N(l_k+1, j-1) \cdot \left(\alpha_U(l_k+1, j-1) \cdot \Pr(N \to U)\right),$$

where

$$p_{01}(l_k, j) = \begin{cases} \beta_N(l_k+1, j-1) \cdot (\alpha_U(l_k+1, j-1) \cdot \Pr(N \to U)), & \text{if } (j - l_k - 1) < \min_{\text{ps}}, \\ \beta_N(l_k+1, j-1) \cdot \alpha_N(l_k+1, j-1), & \text{if } (j - l_k - 1) \geq \min_{\text{ps}}. \end{cases}$$

Accordingly, the probability that there is at least one more accessible base pair in the considered multiloop (i.e., on the structure fragment $R_{(l_k+1)(j-1)}$) is given by $1 - P_{01}^{M_{k+1}}(l_k, j)$.
If no additional accessible base pair is sampled, the sampling process for the considered multibranched loop (closed by pair $r_i.r_j$) is terminated; the resulting loop is thus a $(k+1)$-loop, with k internal helices closed by the k sampled base pairs $r_{h_p}.r_{l_p}$, $1 \leq p \leq k$, accessible from the closing pair $r_i.r_j$. Otherwise, the next accessible base pair $r_{h_{k+1}}.r_{l_{k+1}}$ is sampled and afterwards, it has yet again to be decided whether the loop contains additional accessible base pairs or not (by another "decision" sampling). This process is then repeated until no additional base pair is sampled.

Sm-II.2 Formal Description of the Sampling Process

According to the previous discussion, it should be clear that a secondary structure for a given RNA sequence $r \in \mathcal{L}_r$ of length n can be sampled in the following recursive way: Start with the entire RNA sequence R_{1n} and consecutively compute the adjacent substructures (single-stranded regions and paired substructures) of the exterior loop (from left to right). Any paired substructure, say the kth substructure of the exterior loops, has to be completed by successively folding other loops (hairpins, stacked pairs, bulges, interior and multibranched loops) before the $(k+1)$th adjacent substructure is computed. This means that the folding process performed by the sampling algorithm corresponds to the native folding procedures of RNA molecules (from left to right, due to the aspects of co-transcriptional folding).

Algorithm 3 Sampling an entire secondary structure

```
Require: RNA sequence $r \in \mathcal{L}_r$ of length $n \geq 1$, and
         all previously defined sampling probabilities computed for $r$ (as global variables).
  procedure computeRandomExteriorLoop ($n$)
  $sec = \emptyset$
  Set $i = 1$, $j = n$ and $k = 0$
  while $(j - i + 1) \neq 0$ do
    /*Create $(k+1)$th helix, starting with free base pair $h.l$, $i < h < l < j$, or leave $R_{ij}$ unpaired:*/
    $extLoopType$ = Sample exterior loop substructure type for $R_{ij}$
    if $extLoopType \mathrel{\widehat{=}} P_0^E(i,j)$ /*case (a): $R_{ij}$ is single-stranded:*/ then
      return $sec$
    else if $extLoopType \mathrel{\widehat{=}} P_{ij}^E(i,j)$ /*case (b): $h = i$ and $l = j$:*/ then
      Set $h = i$ and $l = j$
    else if $extLoopType \mathrel{\widehat{=}} \sum_{h=(i+1)}^{(j+1)-\min_{\text{ps}}} P_{hj}^E(i,j,h)$ /*case (c): $i < h < l = j$:*/ then
      Sample $h \in [(i+1), (j+1) - \min_{\text{ps}}]$ according to probabilities $\{P_{hj}^E(i,j,h)\}$
      Set $l = j$
    else if $extLoopType \mathrel{\widehat{=}} \sum_{l=(i-1)+\min_{\text{ps}}}^{(j-1)} P_{il}^E(i,j,l)$ /*case (d): $i = h < l < j$:*/ then
      Set $h = i$
      Sample $l \in [(i-1) + \min_{\text{ps}}, (j-1)]$ according to probabilities $\{P_{il}^E(i,j,l)\}$
    else if $extLoopType \mathrel{\widehat{=}} \sum_{h=(i+1)}^{j-\min_{\text{ps}}} P_{hl}^E(i,j,h)$ /*case (e): $i < h < l < j$:*/ then
      Sample $h \in [(i+1), j - \min_{\text{ps}}]$ according to probabilities $\{P_{hl}^E(i,j,h)\}$
      Sample $l \in [(h-1) + \min_{\text{ps}}, (j-1)]$ according to probabilities $\{\widehat{P}_{hl}^E(j,h,l)\}$
    end if
    /*Collect base pairs for $(k+1)$th substructure and add them to the entire structure:*/
    $sub = \{h.l, (h+1).(l-1), \ldots, (h + (\min_{\text{hel}} - 1)).(l - (\min_{\text{hel}} - 1))\}$
    $sub = sub \cup$ computeRandomLoop $(h + (\min_{\text{hel}} - 1), l - (\min_{\text{hel}} - 1))$
    $sec = sec \cup sub$
    /*Consider the remaining fragment $R_{(l+1)j}$:*/
    Set $i = l + 1$ /*=next unpaired base after free base pair $h.l$*/ and $k = k + 1$
  end while
  return $sec$
  end procedure
```

Algorithm 4 Sampling any substructure of an entire secondary structure

```
procedure computeRandomLoop ($i$, $j$)
Set $sec = \emptyset$
$randLoopType$ = Sample loop type closed by $i.j$
if $randLoopType \widehat{=} Q^{HL}_{ij}(i, j)$ /*$i.j$ closes hairpin loop:*/ then
  return $sec$
else if $randLoopType \widehat{=} Q^{SP}_{ij}(i, j)$ /*$i.j$ closes stacked pair:*/ then
  $sec = sec \cup \{(i+1).(j-1)\}$
  $sec = sec \cup$ computeRandomLoop ($i+1$, $j-1$)
else if $randLoopType \widehat{=} Q^{BI}_{ij}(i, j)$ /*$i.j$ closes bulge or interior loop:*/ then
  $sec = sec \cup$ computeRandomBulgeInteriorLoop ($i$, $j$)
else if $randLoopType \widehat{=} Q^{ML}_{ij}(i, j)$ /*$i.j$ closes multiloop:*/ then
  $sec = sec \cup$ computeRandomMultiLoop ($i$, $j$)
end if
return $sec$
end procedure
```

Algorithm 5 Sampling a bulge or interior loop within a secondary structure

```
procedure computeRandomBulgeInteriorLoop ($i$, $j$)
if Sample strictly corresponding to PF approach then
  /*This requires to use a constant $\max_{BL}$:*/
  Sample $h$ and $l$ according to probabilities $\{P^{BIL}_{hl}(i, j, h, l)\}$
else
  /*This allows $\max_{BL} = \infty$ (then no restrictions are applied):*/
  $loopType$ = Sample bulge or interior loop type for $R_{ij}$
  if $loopType \widehat{=} \sum_{h=(i+2)}^{j-\min_{\text{ps}}} P^{BI}_{hj}(i, j, h)$ /*bulge on the left:*/ then
    Sample $h \in [(i+2), j - \min_{\text{ps}}]$ according to probabilities $\{P^{BI}_{hj}(i, j, h)\}$
    Set $l = j$
  else if $loopType \widehat{=} \sum_{l=i+\min_{\text{ps}}}^{(j-2)} P^{BI}_{il}(i, j, l)$ /*bulge on the right:*/ then
    Set $h = i$
    Sample $l \in [i + \min_{\text{ps}}, (j-2)]$ according to probabilities $\{P^{BI}_{il}(i, j, l)\}$
  else if $loopType \widehat{=} \sum_{h=(i+2)}^{j-\min_{\text{ps}}-1} P^{BI}_{hl}(i, j, h)$ /*interior loop:*/ then
    Sample $h \in [(i+2), j - \min_{\text{ps}} - 1]$ according to probabilities $\{P^{BI}_{hl}(i, j, h)\}$
    Sample $l \in [(h-1) + \min_{\text{ps}}, (j-2)]$ according to probabilities $\{\widehat{P}^{BI}_{hl}(j, h, l)\}$
  end if
end if
$sec = \{h.l, (h+1).(l-1), \ldots, (h + (\min_{\text{hel}} - 1)).(l - (\min_{\text{hel}} - 1))\}$
$sec = sec \cup$ computeRandomLoop ($h + (\min_{\text{hel}} - 1)$, $l - (\min_{\text{hel}} - 1)$)
return $sec$
end procedure
```

For a formal description on how the sampling algorithm works and explicit information on where each of the previously defined sampling probabilities has to be considered in order to perform the needed random choices, see Algorithms 3 to 6.

It remains to mention that when the probabilities $\alpha_x(i, j)$, $x \in \{AT, AB, AO, AN\}$, $1 \leq i, j \leq n$, are also precomputed, each of the needed sampling probabilities can be derived in constant time. Thus, after a preprocessing of the given RNA sequence (which includes the complete inside outside computation and takes cubic time and requires quadratic storage), corresponding secondary structures can be quickly generated. In fact, the time complexity of the sampling algorithm is bounded by $\mathcal{O}(n^2)$, since any structure of size n can have at most $\lfloor \frac{n - \min_{HL}}{2} \rfloor$ base pairs and any base pair can be sampled in linear time.

Algorithm 6 Sampling a multiloop within a secondary structure

```
procedure computeRandomMultiLoop ($i$, $j$)
Set $sec = \emptyset$, $k = 0$ and $l_k = i$
while $(j - l_k - 1) \geq \min_{\text{ps}}$ do
    /*Create $(k+1)$th helix, starting with accessible pair $h_{k+1}.l_{k+1}$, $l_k < h_{k+1} < l_{k+1} < j$:*/
    if $(k+1) = 1$ then
        Sample $h \in [(i+1), j - 2 \cdot \min_{\text{ps}}]$ according to probabilities $\{P^{M_1}_{hl}(i,j,h)\}$
        Sample $l \in [(h-1) + \min_{\text{ps}}, (j-1) - \min_{\text{ps}}]$ according to probabilities $\{\widehat{P}^{M_1}_{hl}(j,h,l)\}$
        Set $h_1 = h$ and $l_1 = l$
    else
        Sample $h \in [(i+1), j - \min_{\text{ps}}]$ according to probabilities $\{P^{M_{k+1}}_{hl}(i,j,h)\}$
        Sample $l \in [(h-1) + \min_{\text{ps}}, (j-1)]$ according to probabilities $\{\widehat{P}^{M_{k+1}}_{hl}(j,h,l)\}$
        Set $h_{k+1} = h$ and $l_{k+1} = l$
    end if
    /*Collect base pairs for $(k+1)$th substructure and add them to the entire structure:*/
    $sub = \{h.l, (h+1).(l-1), \ldots, (h + (\min_{\text{hel}} - 1)).(l - (\min_{\text{hel}} - 1))\}$
    $sub = sub \cup$ computeRandomLoop $(h + (\min_{\text{hel}} - 1), l - (\min_{\text{hel}} - 1))$
    $sec = sec \cup sub$
    /*Decide whether to leave the remaining fragment $R_{(l_{k+1}+1)(j-1)}$ unpaired or not:*/
    if $(k+1) \geq 2$ then
        Sample "decision" according to $P^{M_{k+1}}_{01}(l_{k+1}, j)$ and $1 - P^{M_{k+1}}_{01}(l_{k+1}, j)$
        if $P^{M_{k+1}}_{01}(l_{k+1}, j)$ /*no additional base pairs on $R_{(l_{k+1}+1)(j-1)}$:*/ then
            return $sec$
        else
            Set $k = k + 1$
        end if
    end if
end while
return $sec$
end procedure
```

Sm-III Tables and Figures

CSP_{freq} (selection principle MF struct.):

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0633	0.1216	0.2071	0.2117	0.2639	0.3694
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.2099	0.3699	0.5594	0.5594	0.5599	0.6302
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.2187	0.3833	0.5830	0.5830	0.5835	0.6607
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.2450	0.4448	0.6417	0.6417	0.6422	0.7356
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.2409	0.4364	0.6399	0.6399	0.6403	0.7379

CSP_{freq} (selection principle MEA struct.):

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0416	0.1049	0.1923	0.1960	0.2496	0.3559
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0555	0.2094	0.4193	0.4193	0.4207	0.4679
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0656	0.2446	0.4961	0.4961	0.4984	0.5613
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0772	0.2510	0.4928	0.4928	0.4942	0.5497
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.1008	0.2917	0.5525	0.5525	0.5543	0.6241

CSP_{freq} (selection principle Centroid):

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0264	0.0800	0.1595	0.1627	0.1932	0.2677
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0374	0.1276	0.2973	0.2973	0.2978	0.3130
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0485	0.1623	0.3791	0.3791	0.3800	0.4097
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0536	0.1665	0.3773	0.3773	0.3778	0.4060
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.0758	0.2150	0.4563	0.4563	0.4568	0.5003

Table 13: Results related to the shapes of selected predictions, obtained from our tRNA database (by 10-fold cross-validation procedures, using sample size 1000).

$\mathrm{CSO}_{\mathrm{freq}}$:

Approach	Parameters	Shape Level 0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.5196	0.6740	0.8160	0.8239	0.8798	0.9556
SCFG	$\min_{HL} = 1, \min_{\mathrm{hel}} = 1$	0.6838	0.9459	0.9903	0.9903	0.9908	0.9995
	$\min_{HL} = 1, \min_{\mathrm{hel}} = 2$	0.6806	0.9006	0.9630	0.9635	0.9640	0.9991
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 1$	0.7148	0.9459	0.9875	0.9880	0.9885	0.9991
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 2$	0.7111	0.8997	0.9677	0.9681	0.9686	0.9995

$\mathrm{CS}_{\mathrm{num}}$:

Approach	Parameters	Shape Level 0	1	2	3	4	5
PF	$\max_{BL} = 30$	21.073	58.200	136.67	140.63	205.54	328.56
SCFG	$\min_{HL} = 1, \min_{\mathrm{hel}} = 1$	16.202	98.357	327.26	327.27	327.51	418.80
	$\min_{HL} = 1, \min_{\mathrm{hel}} = 2$	25.205	142.50	453.03	453.03	453.10	527.04
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 1$	24.883	130.04	392.78	392.79	393.05	494.79
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 2$	34.898	173.73	513.05	513.06	513.08	595.26

$\mathrm{DS}_{\mathrm{num}}$:

Approach	Parameters	Shape Level 0	1	2	3	4	5
PF	$\max_{BL} = 30$	355.32	130.22	81.796	33.125	22.585	4.8848
SCFG	$\min_{HL} = 1, \min_{\mathrm{hel}} = 1$	802.27	244.52	60.504	60.030	59.916	28.764
	$\min_{HL} = 1, \min_{\mathrm{hel}} = 2$	652.75	125.69	24.687	24.687	24.687	16.019
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 1$	752.71	208.65	48.257	47.797	47.691	21.838
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 2$	592.84	103.04	18.921	18.921	18.921	12.053

Table 14: Results related to the shapes of sampled structures, obtained from our tRNA database (by 10-fold cross-validation procedures, using sample size 1000).

CSP$_{\text{freq}}$ (selection principle MF struct.):

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0000	0.0009	0.0078	0.0513	0.0261	0.6353
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0000	0.0026	0.0052	0.0131	0.0357	0.7128
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0000	0.0052	0.0139	0.0331	0.0522	0.7502
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0000	0.0044	0.0113	0.0314	0.0766	0.7781
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.0009	0.0096	0.0244	0.0609	0.1027	0.8207

CSP$_{\text{freq}}$ (selection principle MEA struct.):

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0000	0.0052	0.0139	0.0835	0.0696	0.6640
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0000	0.0000	0.0000	0.0000	0.0261	0.3820
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0000	0.0009	0.0009	0.0035	0.0566	0.4769
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0000	0.0000	0.0000	0.0009	0.0261	0.3977
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.0000	0.0009	0.0009	0.0035	0.0557	0.5387

CSP$_{\text{freq}}$ (selection principle Centroid):

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0000	0.0026	0.0104	0.0775	0.0731	0.7214
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0000	0.0000	0.0000	0.0000	0.0104	0.1097
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0000	0.0000	0.0000	0.0000	0.0148	0.1279
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0000	0.0000	0.0000	0.0000	0.0078	0.1236
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.0000	0.0000	0.0000	0.0009	0.0139	0.1549

Table 15: Results related to the shapes of selected predictions, obtained from our 5S rRNA database (by 10-fold cross-validation procedures, using sample size 1000).

CSO_{freq}:

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0009	0.1662	0.3063	0.7580	0.6883	0.9817
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0000	0.2855	0.4526	0.9852	0.9974	1.0000
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0017	0.4135	0.5754	0.9861	0.9983	0.9991
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0000	0.3308	0.4883	0.9904	0.9974	1.0000
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.0026	0.4509	0.6372	0.9904	0.9974	0.9991

CS_{num}:

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0009	0.7571	3.4207	36.641	30.288	600.35
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0000	0.5432	1.1811	20.640	51.834	573.72
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0017	1.1428	2.6615	32.051	64.332	608.06
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0000	0.6651	1.4309	22.983	54.635	569.80
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.0026	1.3795	3.1949	36.673	71.080	609.58

DS_{num}:

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	710.75	333.72	237.71	93.335	63.661	7.0951
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	999.67	941.77	866.98	336.69	167.10	16.476
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	999.18	884.49	764.79	249.02	129.35	14.198
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	999.93	947.19	874.03	331.75	163.09	15.620
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	999.68	885.81	762.67	239.28	123.91	13.558

Table 16: Results related to the shapes of sampled structures, obtained from our 5S rRNA database (by 10-fold cross-validation procedures, using sample size 1000).

CSP$_{\text{freq}}$ (selection principle MF struct.):

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0661	0.1255	0.1586	0.2050	0.2183	0.4834
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0530	0.0993	0.1191	0.1324	0.1589	0.3776
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0398	0.1193	0.1457	0.1656	0.1856	0.4106
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0530	0.1259	0.1390	0.1590	0.1789	0.4107
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.0530	0.1258	0.1522	0.1788	0.1985	0.4240

CSP$_{\text{freq}}$ (selection principle MEA struct.):

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0660	0.1123	0.1453	0.1984	0.2051	0.4902
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0264	0.0927	0.0993	0.1125	0.1325	0.3778
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0264	0.1193	0.1391	0.1523	0.1789	0.4239
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0264	0.0927	0.0993	0.1125	0.1325	0.3777
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.0197	0.1127	0.1391	0.1656	0.2055	0.4109

CSP$_{\text{freq}}$ (selection principle Centroid):

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.0793	0.1321	0.1653	0.1917	0.2449	0.5100
SCFG	$\min_{HL} = 1, \min_{\text{hel}} = 1$	0.0197	0.0861	0.1059	0.1190	0.1258	0.3181
	$\min_{HL} = 1, \min_{\text{hel}} = 2$	0.0197	0.0795	0.0926	0.1191	0.1192	0.3578
	$\min_{HL} = 3, \min_{\text{hel}} = 1$	0.0197	0.0795	0.0926	0.1125	0.1125	0.3181
	$\min_{HL} = 3, \min_{\text{hel}} = 2$	0.0197	0.0927	0.1125	0.1390	0.1391	0.3577

Table 17: Results related to the shapes of selected predictions, obtained from the S-151Rfam database (by 2-fold cross-validation procedures, using sample size 1000).

$\mathrm{CSO}_{\mathrm{freq}}$:

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	0.3638	0.4433	0.4766	0.5231	0.6488	0.7947
SCFG	$\min_{HL} = 1, \min_{\mathrm{hel}} = 1$	0.2520	0.5497	0.6095	0.6888	0.7683	0.9604
	$\min_{HL} = 1, \min_{\mathrm{hel}} = 2$	0.2717	0.5630	0.6158	0.7284	0.8079	0.9605
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 1$	0.2518	0.5429	0.6093	0.7218	0.7815	0.9472
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 2$	0.2715	0.5564	0.6027	0.7087	0.7484	0.9604

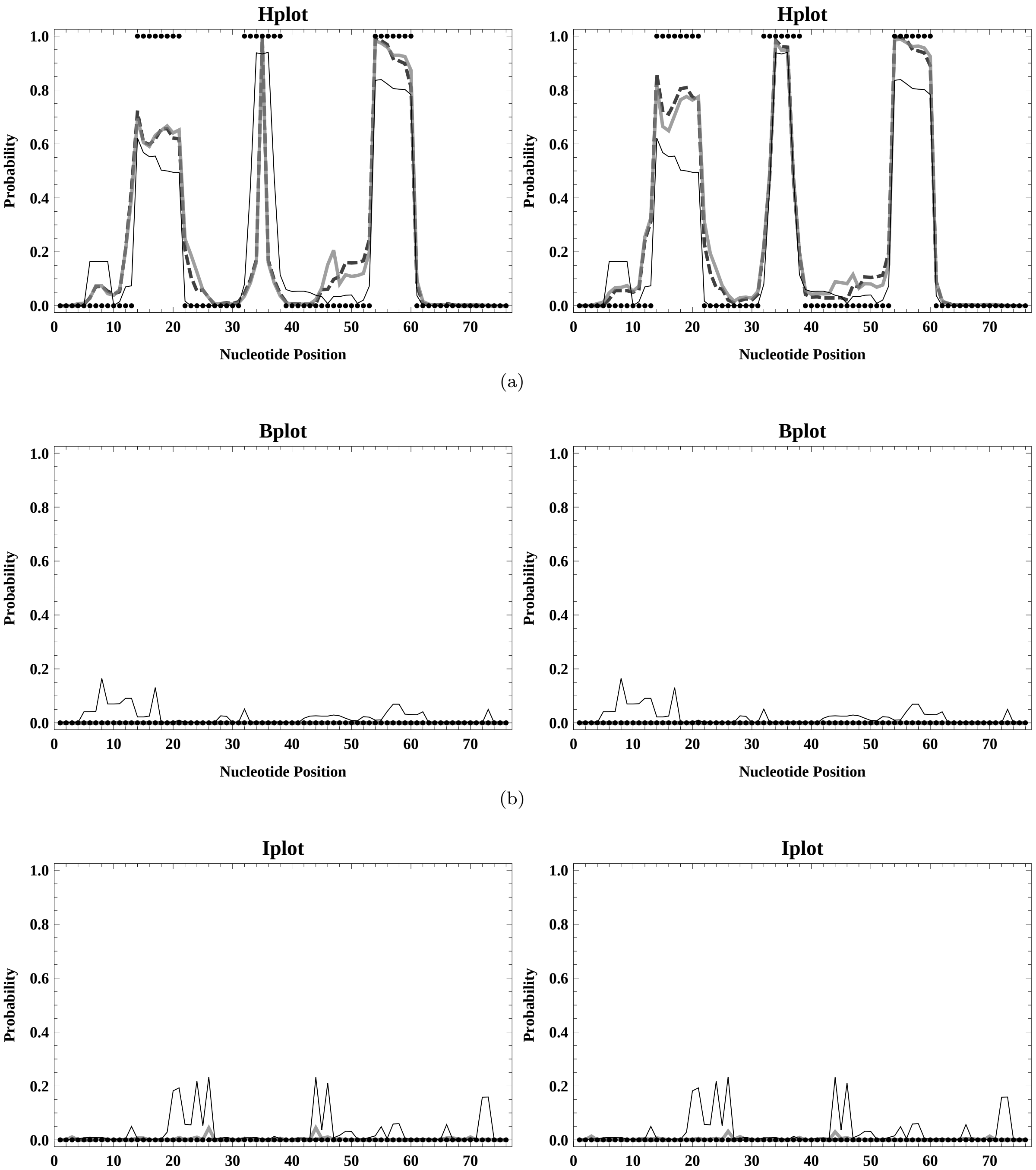
$\mathrm{CS}_{\mathrm{num}}$:

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	40.390	88.886	121.55	158.32	195.83	453.58
SCFG	$\min_{HL} = 1, \min_{\mathrm{hel}} = 1$	10.743	47.281	63.587	97.088	121.64	362.44
	$\min_{HL} = 1, \min_{\mathrm{hel}} = 2$	12.968	58.796	78.776	115.96	139.09	387.16
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 1$	12.468	51.569	67.603	104.67	125.50	365.84
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 2$	15.059	63.707	83.965	125.82	142.99	391.39

$\mathrm{DS}_{\mathrm{num}}$:

Approach	Parameters	Shape Level					
		0	1	2	3	4	5
PF	$\max_{BL} = 30$	540.74	304.36	255.40	150.89	117.24	18.795
SCFG	$\min_{HL} = 1, \min_{\mathrm{hel}} = 1$	892.14	600.39	526.36	368.49	322.88	99.601
	$\min_{HL} = 1, \min_{\mathrm{hel}} = 2$	849.32	538.56	466.17	322.99	286.12	84.480
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 1$	888.89	588.97	516.66	358.72	315.25	94.603
	$\min_{HL} = 3, \min_{\mathrm{hel}} = 2$	840.03	522.53	452.04	307.61	273.92	77.536

Table 18: Results related to the shapes of sampled structures, obtained from the S-151Rfam database (by 2-fold cross-validation procedures, using sample size 1000).

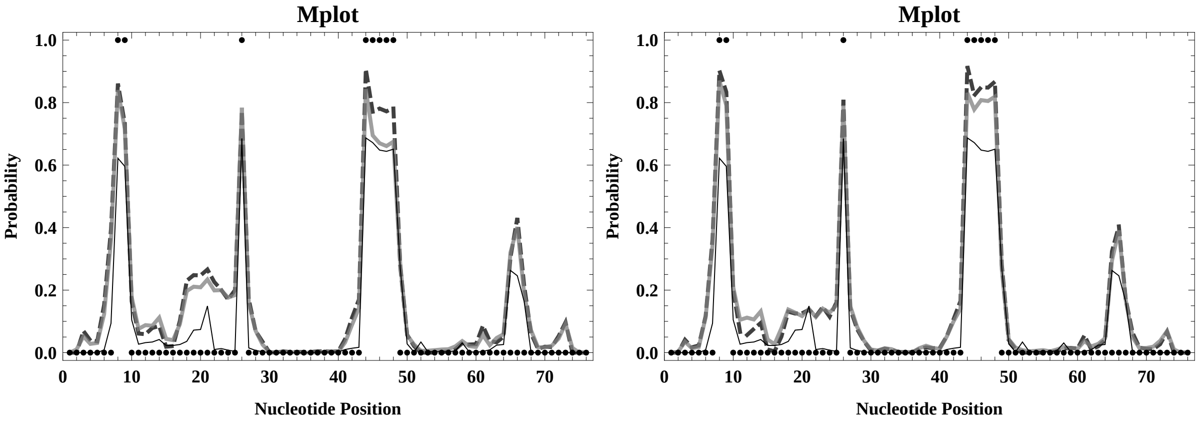

Hplot
Hplot
Bplot
Bplot
Iplot
Iplot
Probability
Nucleotide Position
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
50
60
70

(a)

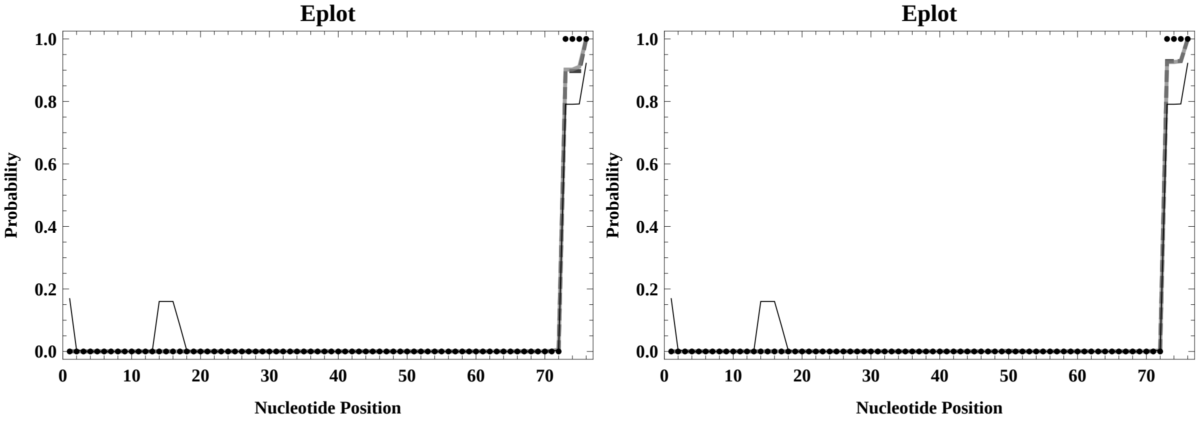
(b)

(c)

Figure 2

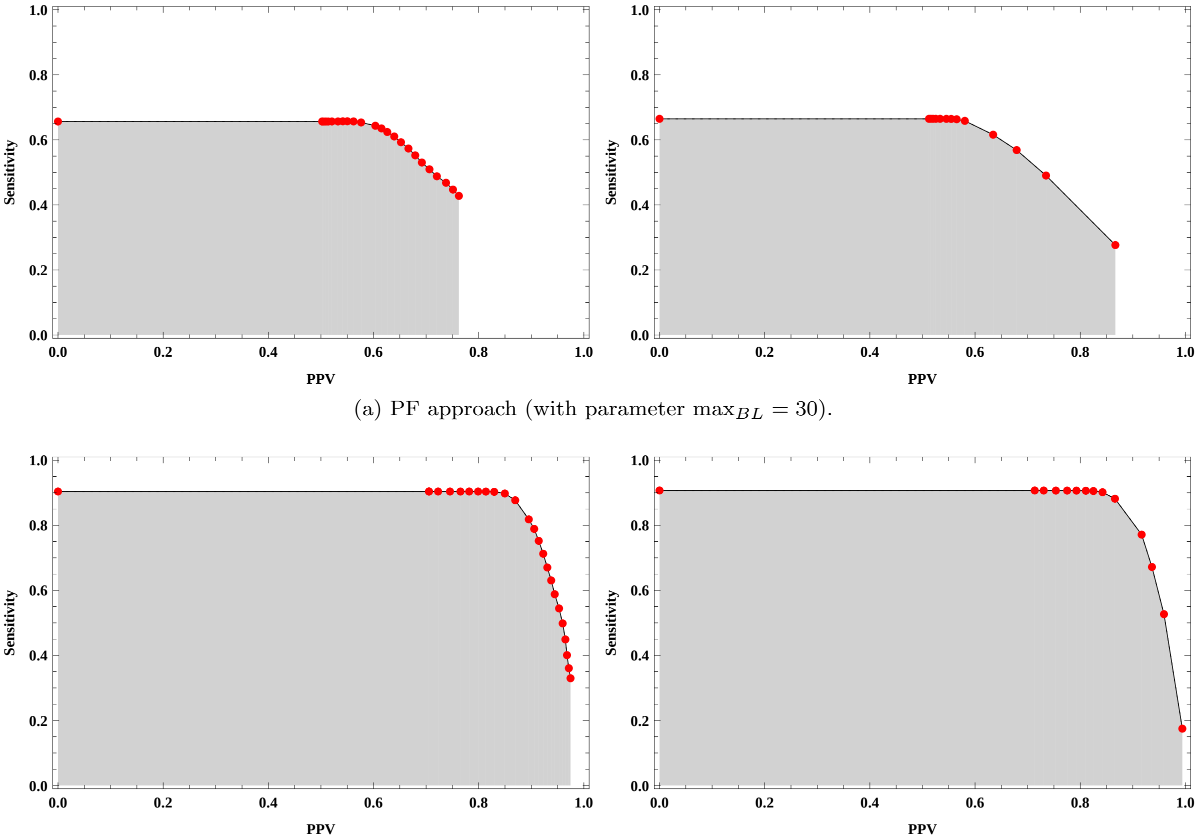


(d)



(e)

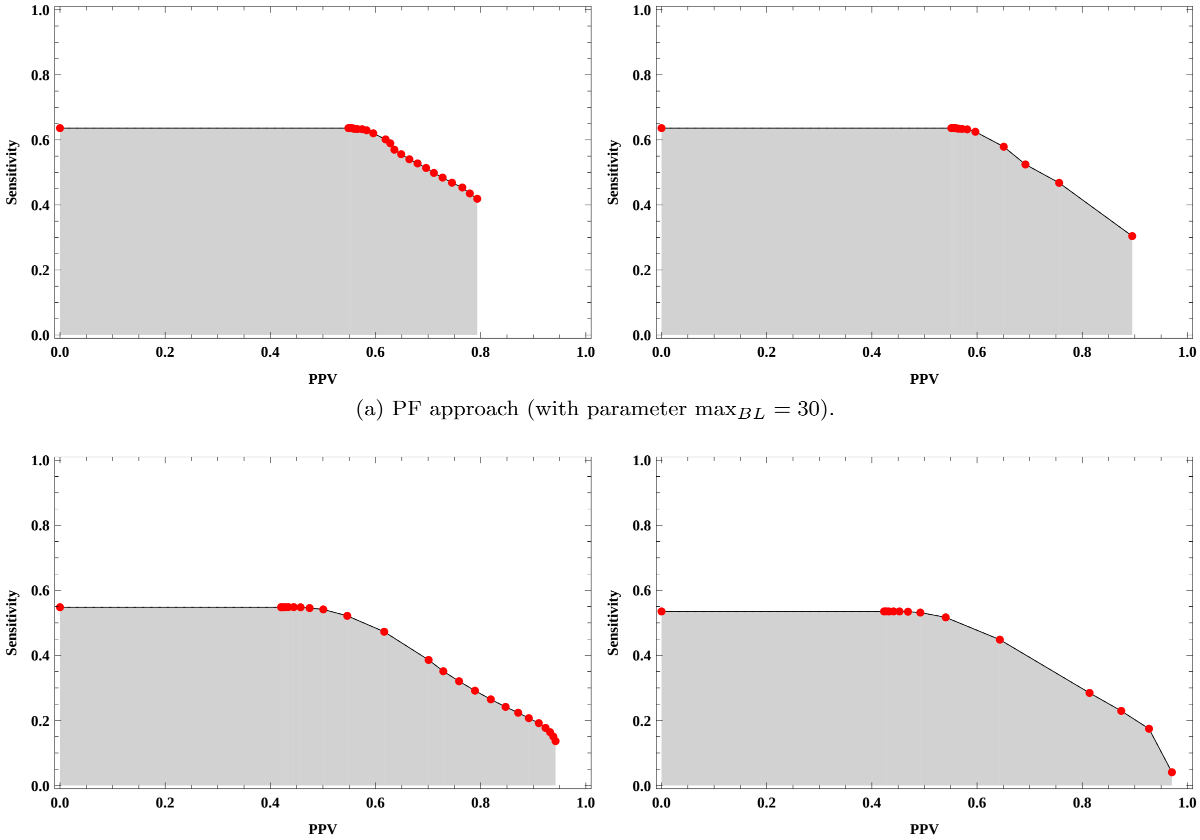
Figure 2: Comparison of loop profiles for *E.coli* tRNA^{Ala}. Hplot, Bplot, Iplot, Mplot and Extplot display the probability that an unpaired base lies in a hairpin, bulge, interior, multibranched and exterior loop, respectively. For each considered variant, these five probabilities are computed by a sample of 1000 structures. Results for the PF approach (for $\max_{BL} = 30$) are displayed by the thin black lines. For the SCFG approach, we chose $\min_{\text{hel}} = 1$ (thick gray lines) and $\min_{\text{hel}} = 2$ (thick dashed darker gray lines), combined with $\min_{HL} = 1$ (figures shown on the left) and $\min_{HL} = 3$ (figures on the right), respectively. The corresponding probabilities for the correct structure of *E.coli* tRNA^{Ala} are also displayed (by black points).



(a) PF approach (with parameter $\max_{BL} = 30$).

(b) SCFG approach (with the most realistic parameter combination $\min_{HL} = 3$ and $\min_{\text{hel}} = 2$).

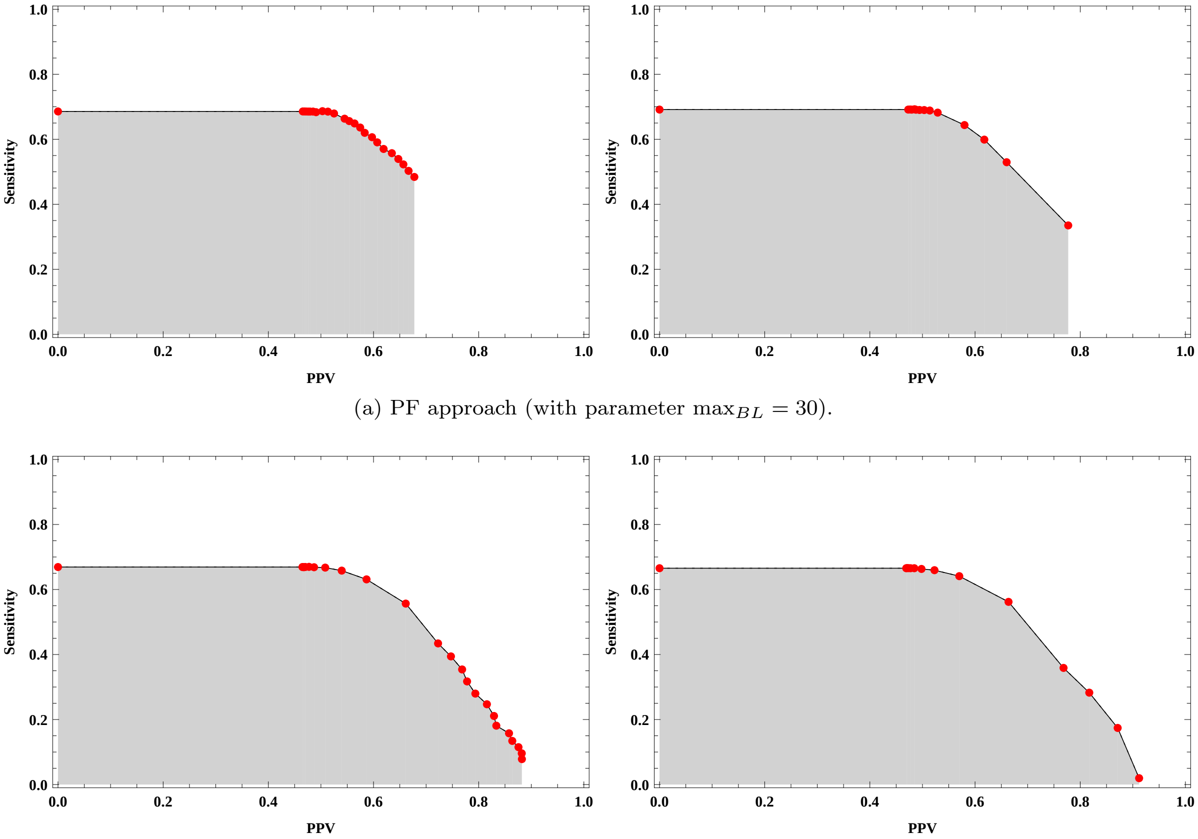
Figure 3: Comparison of the (areas under) ROC curves obtained for our tRNA database (computed by 10-fold cross-validation procedures, using sample size 1000). For each considered sampling variant, the corresponding ROC curves are shown for prediction principle MEA structure (figure on the left) and centroid (figure on the right), respectively.



(a) PF approach (with parameter $\max_{BL} = 30$).

(b) SCFG approach (with the most realistic parameter combination $\min_{HL} = 3$ and $\min_{\text{hel}} = 2$).

Figure 4: Comparison of the (areas under) ROC curves obtained for our 5SrRNA database (computed by 10-fold cross-validation procedures, using sample size 1000). For each considered sampling variant, the corresponding ROC curves are shown for prediction principle MEA structure (figure on the left) and centroid (figure on the right), respectively.



(a) PF approach (with parameter $\max_{BL} = 30$).

(b) SCFG approach (with the most realistic parameter combination $\min_{HL} = 3$ and $\min_{\text{hel}} = 2$).

Figure 5: Comparison of the (areas under) ROC curves obtained for the mixed S-151Rfam database (computed by two-fold cross-validation procedures, using the same folds as in [DWB06] and sample size 1000). For each considered sampling variant, the corresponding ROC curves are shown for prediction principle MEA structure (figure on the left) and centroid (figure on the right), respectively.