Non-Redundant Sampling and Statistical Estimators for RNA Structural Properties at the Thermodynamic Equilibrium

Christelle Rovetta, Juraj Michálik, Ronny Lorenz, Andrea Tanzer, Yann Ponty

Non-Redundant Sampling and Statistical Estimators for RNA Structural Properties at the Thermodynamic Equilibrium

Christelle Rovetta*, Juraj Michalik*, Ronny Lorenz, Andrea Tanzer, and Yann Ponty,

Abstract—The computation of statistical properties of RNA structure at the thermodynamic equilibrium, or Boltzmann ensemble of low free-energy, represents an essential step to understand and harness the selective pressure weighing on RNA evolution. However, classic methods for sampling representative conformations are frequently crippled by large levels of redundancy, which are uninformative and detrimental to downstream analyses.
In this work we introduce, and implement within the Vienna RNA package, a highly-efficient non-redundant backtracking procedure to produce collections of unique secondary structures generated within a well-defined distribution. This procedure is coupled with a novel statistical estimator, which we prove is unbiased, consistent and has lower variance (better convergence) than the classic estimator.
We show the efficiency of our coupled non-redundant sampler/estimator by revisiting several applications of sampling in RNA bioinformatics, and by demonstrating its practical superiority over previous estimators. We conclude by discussing the choice of the number of samples required to produce reliable estimates.

Index Terms—RNA secondary structure, Boltzmann equilibrium, Non-redundant sampling, Statistical estimator

1 Introduction

Structural properties of RNAs are crucial to build a mechanical understanding of their function. Aside from their role in mediating genetic information from genome to the protein levels, RNAs are associated with multiple enzymatic and regulatory functions, leading recent versions of the RFAM database to enumerate more than 3,000 functional families [1]. This collection will likely expand in the upcoming years, following the discovery of hundreds of thousands of long non-coding RNAs [2]. In many of those families, an evolutionary pressure can be observed towards the adoption of one or several important folds, leading to an instrumental part being played by a consensus secondary structure in the definition of functional families.

RNA functional architectures are adopted as the final outcome of a folding process governed by thermodynamics. Multiple copies of an RNA alternate between their stable structures, inducing an equilibrium which favors a subset of stables conformations, the Boltzmann ensemble of low-energy including the minimum free-energy structure. The concept of Boltzmann ensemble is found at the core of recent computational methods, allowing to embrace the full conformational diversity. Following the seminal work of McCaskill [3], essential thermodynamics quantities, namely the partition function and base-pairs probabilities, can be computed in polynomial time using dynamic programming (DP). Modified versions of the McCaskill DP scheme have been proposed over the past decades to compute other properties, including the expected 5′–3′ distance [4], base-pair distance [5], [6] or mutation-classified [7] partition functions, moments of the free-energy distribution [8] and of general additive features [9]. However, more complex quantities, such as the graph distance distribution, may require algorithms whose complexity, albeit polynomial, become prohibitively large [10]. Moreover, the study of new statistical quantities of interest requires the design and implementation of new, sometimes highly involved, algorithmic DP schemes, which are the object of the work.

As an alternative, statistical approaches are increasingly used to estimate features of the Boltzmann ensemble. First introduced by Ding and Lawrence [11], a stochastic backtrack procedure samples RNA secondary structures from the exact Boltzmann distribution by recursively performing local random choices, using precomputed probabilities. Boltzmann sampling procedures are now implemented in most libraries for RNA secondary structure analysis, including the `ViennaRNA` package [12], `RNAStructure` [13] and `Unafold` [14]. Such approaches are also used to sample from reduced subsets of secondary structures sharing a certain property, such as locally optimal structures [15], [16], or within partitioned sets [17]. Sampling methods possess a wide range of applications, including RNA kinetics studies [18], evolutionary neutrality [19], structure modeling from experimental probing data [20], gradient-based optimization strategies [21], and RNA design [22]. By only requiring a capacity to compute the quantity of interest, such methods represent a flexible, if approximate, alternative to exact DP-based computations. Sampling may even represent the only available option for the production of dominant conformers in coarse-grained abstractions [23].

* *Both authors contributed equally*

- *Y. Ponty, J. Michalik and C. Rovetta were with the Department of Computer Science (LIX), Ecole Polytechnique, 91 120 Palaiseau, France. E-mail: yann.ponty@lix.polytechnique.fr*
- *R. Lorenz and A. Tanzer were with the Theoretical Biochemistry Institute, University of Vienna, Austria.*

Manuscript received XXXX; revised YYYY.

Statistical estimates for the Boltzmann ensemble are classically computed by first generating a fixed number of structures, followed by an evaluation of features of interest. Redundancy within the sample, *i.e.* the presence of multiple copies of the same conformation, is typically used to estimate emission probabilities. However, in Boltzmann-Gibbs distributions, the exact probability of emission for any given structure only depends on its free-energy, renormalized by the partition function, both of which are readily available after each generation. Redundancy is therefore uninformative, since estimated probabilities are detrimental to the accuracy of derived.

Moreover, a high level of redundancy within sampled sets in Boltzmann-like distributions is theoretically expected [24], [25], leading one to anticipate a substantial impact of redundancy on performances. We previously introduced general principles for the non-redundant generation of non-redundant sets [26], [16]. However, while the overall sample distribution remained well-defined, these work left open the computation of statistical estimates from non-redundant samples. Computing statistics from such a non-redundant sample is not a trivial task since non-redundant sampling induces dependency between subsequent generation, thereby breaking the assumption of independence between consecutive generations. Such a dependency ultimately biases the output of naive estimators.

In this work, we address both the random generation of non-redundant subset of RNA structures, and the computation of statistical estimates from non-redundant subsets. Our main contributions include:

- The first polynomial-time algorithm for the non-redundant Boltzmann generation of RNA structures in the realistic Turner energy model, generalizing the statistical sampling algorithm introduced by Ding and Lawrence [11];
- The implementation of our algorithm in the popular `ViennaRNA` package [12];
- A novel statistical estimator from non-redundant sample, which is unbiased, asymptotically converges, and has lower empirical variance than the naive estimator for the same sample set;
- Extensive empirical analysis demonstrates that our non-redundant sampler, coupled with a corrected estimator, enables more precise estimates for RNA structural features.

Typical estimates of interest include base-pair probabilities matrices (dot-plots), shape probabilities [23] and graph distance distribution [10], for which exact dynamic-programming alternatives are prohibitively costly. We finally discuss the number of samples required to achieve a given precision for the estimates.

2 Material and Methods

An RNA can be abstracted as a sequence of nucleotides $w \in \{\mathsf{A}, \mathsf{C}, \mathsf{G}, \mathsf{U}\}^\star$, having size $|w| = n$. A **base pair** is a pair of positions $(i, j) \in [1, n]^2$ such that $i < j$, and is **valid** if its content $\{w[i], w[j]\}$ is either $\{\mathsf{A}, \mathsf{U}\}, \{\mathsf{C}, \mathsf{G}\}$, or $\{\mathsf{G}, \mathsf{U}\}$. Base pairs represent the formation of hydrogen bonds between nucleotides, and participate in the stability of individual conformations. In the following, we denote by $\mathcal{P}$ the set of all **valid base pairs** in an RNA w.

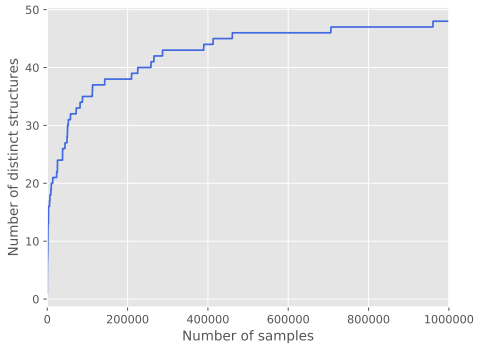


Fig. 1. **Distinct structures within a redundant sample.** Generation of 10^6 Boltzmann-distributed structures for a small RNA `GGCGGAACCGUC`. The number of distinct structures quickly reaches a plateau after 170 000 generations, and only 48 out of the possible 84 possible structures are represented in the final output.

A **secondary structure** $s \subseteq \mathcal{P}$ is set of valid base pairs satisfying the following constraints:

1) No crossing interactions: $\forall (i, j) \neq (k, l) \in s$ such that $i < k$, *i.e.* either $i < k < l < j$ ((k, l) nested within (i, j)) or $i < j < k < l$ ((i, j) precedes (k, l));
2) Any base can participate in at most one base pair within the same secondary structure (monogamy).

For instance, the secondary structure S_1 in Figure 1, can be represented as a set of base pairs $\{(1, 12), (2, 11), (3, 10), (4, 9)\}$. In the following, we use Ω to denote the set of **all valid secondary structures** for the input RNA w.

An **energy model** associates a free-energy $E(s) \in \mathbb{R}$ to any structure $s \in \Omega$ for an RNA w. This allows to define the **Boltzmann factor** $\mathcal{B}(s)$ of s as

$$\mathcal{B}(s) = e^{-\beta E(s)}$$

s where $\beta := 1/RT$, R is the Boltzmann constant, and T is the temperature, both expressed in coherent units. Boltzmann factors are essential thermodynamic quantities, that can be used to describe the behavior of a system at steady-state distribution.

Indeed, at the **thermodynamic equilibrium**, the structures in Ω are expected to follow a **Boltzmann distribution**, *i.e.* each structure is observed with probability proportional to its Boltzmann factor. More precisely, the **Boltzmann probability** $\mathbb{P}(s)$ of any given conformation $s \in \Omega$ is defined as

$$\mathbb{P}(s) := \frac{\mathcal{B}(s)}{\mathcal{Z}}$$

where $\mathcal{Z}$ is the **partition function** of w, defined as

$$\mathcal{Z} = \sum_{s \in \Omega} \mathcal{B}(s).$$

The partition function not only acts as a normalization factor, but many useful quantities can be derived from its various derivatives and constrained values.

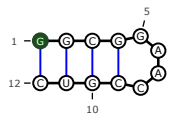
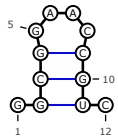
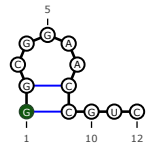
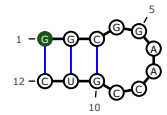
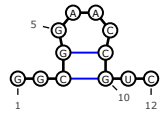
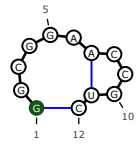
Id	Structure	Bolz. Prob.	$\mathbb{P}$(Abs.)	$\mathbb{E}$ (#Occ)	#Occ.
S_1		80.8 %	1.12 10^{-36}	40.44	35
S_2		9.81 %	0.57 %	4.91	8
S_3		3.71 %	15.1 %	1.85	3
S_4		2.28 %	31.6 %	1.14	1
S_5		1.93 %	37.6 %	0.969	2
S_6		0.73 %	69.3 %	0.366	1
⋮		⋮	⋮	⋮	⋮
S_{84}		9.01 10^{-10}	100%	0.0	0

TABLE 1
Expected and observed redundancy within a (redundant) random sample of 50 structures, generated for the RNA GGCGGAACCGUC. Only 6 out of the 84 possible conformations are represented in the redundant sample (expected value = 4.76).

2.1 The origin and challenge of redundancy

For a given RNA sequence, the free energies of compatible secondary structures greatly vary. Thus, in the matching Boltzmann distribution, probabilities associated with secondary structures cover a wide range of values. In particular, it is not uncommon for a small subset of structures to accumulate most of the probability mass, and therefore be recurrently represented in statistical samples.

Figure 1 illustrates the practical impact of redundancy, and reveals the abysmally-slow emergence of novel structures while generating 10^6 random/Boltzmann distributed structures. Figure 1 further describes the Boltzmman ensemble of our running example, and provides a theoretical explanation for the number of distinct structures to grow at a painstakingly slow pace: While sampling structures, the first collision (*i.e.* first duplicate in a dataset) is expected after only 3.5 generations [24], leading to an expected 4.75 distinct structures within a sample of 50 structures, only increasing to 9 structures for 10^3 structures, and finally 15 structures for 10^6 structures. Generating the full collection (84 secondary structures) is expected to require between $1.09 \cdot 10^9$ and $6.8 \cdot 10^{10}$ structures, using standard theoretical estimates [24]

To mitigate the useless wastefulness of redundancy, one could perform **non-redundant sampling**, avoiding previously-generated structures in subsequent generations. Denote by Θ the set of structures previously generated at a given step of the algorithm, then the probability of emitting a new structure $t \in \Omega \setminus \Theta$ is given by:

$$\mathbb{P}_\Theta(t) = \begin{cases} \frac{\mathbb{P}(t)}{1-\mathrm{Cov}(\Theta)} & \text{if } s \notin \Theta \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $\mathrm{Cov}(\Theta)$ is the **coverage** of Θ, *i.e.* the cumulated probability of a set Θ of distinct structures:

$$\mathrm{Cov}(\Theta) = \sum_{s \in \Theta} \mathbb{P}(s)$$

The overall probability to generate an ordered non-redundant collection of m structures $\mathbf{t} := (t_1, t_2, \ldots, t_m)$ is thus:

$$\mathbb{P}(\mathbf{t}) = \mathbb{P}(t_1) \frac{\mathbb{P}(t_2)}{1-\mathbb{P}(t_1)} \cdots \frac{\mathbb{P}(t_m)}{1-\mathbb{P}(t_1)-\cdots-\mathbb{P}(t_{m-1})}. \quad (2)$$

2.2 Revisiting the classic (redundant) statistical sampling

Before introducing our non-redundant alternative, let us remind the principles underlying the classic (redundant) statistical sampling of RNA structures. Secondary structures compatible with w can be randomly generated in a Boltzmann distribution, using a **stochastic backtrack** introduced by Ding and Lawrence [11]. For the sake of simplicity, we illustrate our general approach, using a base-pair based energy model [27], where any base pair (i, j) has associated contribution $E_{i,j}$. The underlying principle easily generalizes to the loop-based Turner model [28], also supported by our implementation.

2.2.1 Dynamic programming as a sequential generative scheme

Following strategies pioneered by Waterman [29] and Nussinov/Jacobson [27], the set of all secondary structures Ω can be recursively defined. Consider $w_{i,j}$, the subsequence of w restricted to the interval $[i, j] \subseteq [1, n]$ ($w_{1,n} \equiv w$), and let $\Omega_{i,j}$ be the set of all secondary structures compatible with $w_{i,j}$. Then we have

$$\Omega_{i,j} := \Omega_{i+1,j} \cup \bigcup_{\substack{(i,k)\in\mathcal{P} \\ \text{s.t. } i<k\leq j}} \{(i,k)\} \times \Omega_{i+1,k-1} \times \Omega_{k+1,j}. \quad (3)$$

The first term represents structures leaving i unpaired, and the second one covers all possible partners for i. The set of all secondary structures is then given by $\Omega := \Omega_{1,n}$.

The construction described in Equation 3 can then be used to define the **tree of valid structures**, also illustrated by Figure 2. In this tree, each node is indexed by a couple (I, P) where I **is a sequence of non-overlapping intervals**, each a subset of $[1, n]$, and $P \subseteq \mathcal{P}$ **a set of pairwise non-crossing base pairs**. We distinguish two types of nodes: leaves and internal nodes. A leaf is a node $(\varnothing, P)$, where P represents a single secondary structure. Conversely, an internal node (I, P) represents the set of secondary structures, consisting

of the base pairs in P, augmented by base pairs chosen for each of the intervals in I. The **root** of the tree is the node $v_{\mathrm{r}} := ([1,n],\varnothing)$, and is associated with the complete set $\Omega_{1,n}$ of all compatible secondary structures, as defined in Equation (3).

The tree is constructed from the root to the leaves using Equation (3). The **children** of an internal node $v := ([i,j] \cup I', P)$ are defined as follows:

$$\mathcal{C}(v) := \{([i+1,j].I', P)\} \bigcup_{\substack{i<k\leq j \\ \text{s. t. } (i,k)\in\mathcal{P}}} \{([i+1,k-1]\cup[k+1,j]\cup I', P\cup\{(i,k)\})\}.$$

We can thus define the **partition function** of a node. It is computed recursively from the leaves to the root as follow:

$$\mathcal{B}((I,P)) = \begin{cases} \mathcal{B}(P) & \text{if } I=\varnothing \text{ (leaf)} \\ \sum_{v'\in\mathcal{C}((I,P))} \mathcal{B}(v') & \text{otherwise.} \end{cases}$$

2.2.2 Sequential stochastic backtrack as a random walk

To randomly generate a secondary structure in the associated distribution, we perform a random walk within the tree, starting at the root v_{r}. At each step of the generation, a child v' of a node v is chosen with probability

$$p_{v\to v'} := \frac{\mathcal{B}(v')}{\mathcal{B}(v)}. \tag{4}$$

The process is iterated until a leaf $(\varnothing, P)$ is reached, and the secondary structure P is then returned. The emission probability for given structure s – to reach a leaf $(\varnothing, s)$ – is then given by:

$$\mathbb{P}(s) = \frac{\mathcal{B}(v_1)}{\mathcal{B}(v_{\mathrm{r}})} \times \frac{\mathcal{B}(v_2)}{\mathcal{B}(v_1)} \times \cdots \times \frac{\mathcal{B}(v_p)}{\mathcal{B}(v_{p-1})} \times \frac{\mathcal{B}(s)}{\mathcal{B}(v_p)} = \frac{\mathcal{B}(s)}{\mathcal{Z}}$$

where $v_{\mathrm{r}} =: v_0, v_1, \ldots, v_p$ is the sequence of nodes encountered along the random walk from the root to the leaf. Thus choosing one of the local alternatives with probability proportional to its contribution, as depicted by Equation (4), is sufficient to ensure that the global emission probabilities fully respect the targeted Boltzmann distribution.

Of course, since the number of secondary structures grows exponentially with the length of the sequence [30], one cannot realistically memorize the entirety of such a tree. However, the additivity of energy models for RNA structure induce a – much simplified – expression for the partition function associated with a node (I,P). Indeed, we have:

$$\mathcal{B}((I,P)) = \mathcal{B}(P) \prod_{[i,j]\in I} \mathcal{B}_{i,j} \tag{5}$$

where $\mathcal{B}_{i,j}$ corresponds to the partition function of all structures confined to an interval $[i,j]$ computable in $\mathcal{O}(n^3)$ time using dynamic programming through:

$$\mathcal{B}_{i,j} = \mathcal{B}_{i+1,j} + \sum_{i<k\leq j} \mathcal{B}((i,k)) \times \mathcal{B}_{i+1,k-1} \times \mathcal{B}_{k+1,j}. \tag{6}$$

Interestingly, injecting the expression of Equation (5) in the transition probability substantially simplifies the transition probabilities. For instance, in the context of a node

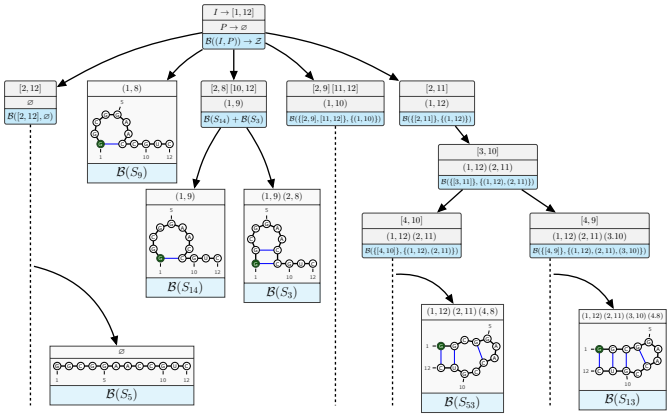


Fig. 2. **Tree of valid secondary structures** (truncated) for the RNA sequence GGCGGAACCGUC. Classic stochastic backtrack can be seen as the generation of a random path from the root to a leaf, at each step choosing one of the local alternatives with probability proportional to their partition function. The emissions of S_3 results from two consecutive choices, and occurs with probability $\frac{\mathcal{B}(S_{14})+\mathcal{B}(S_3)}{\mathcal{Z}} \frac{\mathcal{B}(S_3)}{\mathcal{B}(S_{14})+\mathcal{B}(S_3)} = \frac{\mathcal{B}(S_3)}{\mathcal{Z}}$, consistent with the targeted Boltzmann distribution.

$v := (I := [i,j]\cdot I', P)$, selecting a base pair (i,k) leads to a new node

$$v' := ([i+1,k-1]\cdot[k+1,j]\cdot I', P\cup\{(i,k)\}),$$

and we get

$$\begin{aligned} p_{v\to v'} :=& \frac{\mathcal{B}(v')}{\mathcal{B}(v)} \\ =& \frac{\mathcal{B}((i,k))\,\mathcal{B}(P)\,\mathcal{B}_{i+1,k-1}\,\mathcal{B}_{k+1,j}\,\prod_{[a',b']\in I'}\mathcal{B}_{a',b'}}{\mathcal{B}(P)\,\mathcal{B}_{i,j}\,\prod_{[a,b]\in I'}\mathcal{B}_{a,b}} \\ =& \frac{\mathcal{B}((i,k))\,\mathcal{B}_{i+1,k-1}\mathcal{B}_{k+1,j}}{\mathcal{B}_{i,j}}. \end{aligned}$$

Remark that the set of previously assigned base pairs no longer influences the probability of choosing a given alternative, and that all of the terms are accessible after a $\mathcal{O}(n^3)$ preprocessing (*c.f.* Equation 6). This crucial observation is at the core of the efficient $\mathcal{O}(n^2)$ stochastic backtrack procedure of Ding and Lawrence [11], further reducible to $\mathcal{O}(n\sqrt{n})$ [31]. Unfortunately, this simplification no longer holds when a set Θ of structures needs to be avoided. Indeed, the above probabilities then need to account for structures in Θ in a way that depends on the structures previously generated, calling for an alternative strategy.

2.3 Sequential non-redundant (NR) sampling

Non-redundant sampling can be reduced to a random generation of a Boltzmann-distributed secondary structure avoiding a predefined set Θ of forbidden structures (see Equation (1)). Remark that Equation (4) induces a Boltzmann distribution as soon as the individual partition functions are correctly computed. A similar strategy could thus be used, using partition functions that reflect the unavailability of structures from Θ, to generate a structure with respect to the distribution defined by Equation (1).

In earlier work dedicated to context free-languages [9] and local minima of RNA kinetics landscapes [16], we

introduced a dedicated data structure $\widetilde{\mathcal{B}}$ to gather and access the contributions of a set of structures forbidden during the stochastic backtrack. Built similarly as the graph of secondary structures, it is restricted to structures in Θ, and thus grants access to the partition function $\widetilde{\mathcal{B}}(v)$ of structures from Θ that were – but should no longer be – generated from a given node v.

2.3.1 General strategy and correctness

In practice, during the stochastic backtrack phase, we modify the probability of choosing some admissible child $v' \in \mathcal{C}(v)$ so that

$$\tilde{p}_{v \to v'} := \frac{\mathcal{B}(v') - \widetilde{\mathcal{B}}(v')}{\mathcal{B}(v) - \widetilde{\mathcal{B}}(v)}.$$

The probability of generating a structure s becomes

$$\mathbb{P}(s) = \frac{\mathcal{B}(v_1) - \widetilde{\mathcal{B}}(v_1)}{\mathcal{Z} - \mathcal{B}(\Theta)} \frac{\mathcal{B}(v_2) - \widetilde{\mathcal{B}}(v_2)}{\mathcal{B}(v_1) - \widetilde{\mathcal{B}}(v_1)} \cdots \frac{\mathcal{B}(s)}{\mathcal{B}(v_{p-1}) - \widetilde{\mathcal{B}}(v_{p-1})}.$$

By repeatedly cancelling adjacent pairs of numerators and denominators, we are left with

$$\mathbb{P}(s) = \frac{\mathcal{B}(s)}{\mathcal{Z} - \mathcal{B}(\Theta)} = \frac{\mathbb{P}(s)}{1 - \sum_{s' \in \Theta} \mathbb{P}(s')}$$

in which one recognizes our targeted distribution, stated in Equation (1). In order to efficiently access all relevant $\widetilde{\mathcal{B}}(v')$ terms, we rely on a specialized data structure further described in the next section.

Algorithm 1: Non-redundant stochastic backtrack.

Data: Forbidden structures $\Theta \subset \Omega$, preprocessing into data structure $\widetilde{\mathcal{B}}$
Result: $s \in \Omega - \Theta$, randomly generated with probability proportional to $\mathcal{B}(s)$; Updated data structure $\widetilde{\mathcal{B}}$ reflecting $\Theta' := \Theta \cup \{s\}$

$v \leftarrow ([1, n], \varnothing)$; `// Start from root` v_r `of` T
$L \leftarrow [v]$; `// Remembers choices to update` $\widetilde{\mathcal{B}}$
while *v is not a leaf* **do**
  Choose $v' \in \mathcal{C}(v)$ with probability $\frac{\mathcal{B}(v') - \widetilde{\mathcal{B}}(v')}{\mathcal{B}(v) - \widetilde{\mathcal{B}}(v)}$;
  $L \leftarrow L + v'$;
  $v \leftarrow v'$;
$(\varnothing, s) \leftarrow v$; `//` v `is a leaf, i.e. a structure` s
for $v' \in L$ **do**
  $\widetilde{\mathcal{B}}(v') \leftarrow \widetilde{\mathcal{B}}(v') + \mathcal{B}(s)$; `// Update` $\widetilde{\mathcal{B}}$ `to add` s
return $s, \widetilde{\mathcal{B}}$

2.3.2 An efficient data structure for $\widetilde{\mathcal{B}}$

For our Nussinov-style stochastic backtrack, the data structure is concretely implemented as a tree such that:

- Each leaf l implicitly represents an individual structure s, and is adjoined a field storing the Boltzmann weight of l, *i.e.* $\widetilde{\mathcal{B}}(l) := \mathcal{B}(s)$;
- Each internal node v is implicitly associated with a pair $(I, P) := ([i, j] \cdots, P)$ defined as above.
  Each edge to a child node v' is explicitly labeled with a field $\xi(x \to v')$ whose content characterize one of left-to-right transitions available from $[i, j]$ in the DP scheme, *i.e.* either $\xi(v \to v') := k$ if a base pair (i, k) is formed, or $\xi(v \to v') := i$ if i is left unpaired.
  Finally, we add a field $\widetilde{\mathcal{B}}(v)$ storing the Boltzmann weight of all structures/leaves within the current subtree:

$$\widetilde{\mathcal{B}}(v) := \sum_{s \text{ in subtree}(v)} \mathcal{B}(s) = \sum_{v' \text{child of } v} \widetilde{\mathcal{B}}(v')$$

The stochastic backtrack is then adapted to navigate the data structure while alternatives (base paired/unpaired fate for first undetermined position) are being investigated/validated, so that:

1) At any given time, the current node v corresponds to the (I, P) pair induced by prior backtrack choices. This invariant can be preserved, following a choice (i, k) (possibly $k = i$ if i left unpaired), by performing a transition $v \to v'$ such that $\xi(v \to v') = k$;
2) Anytime a node is visited by the backtrack, its $\Theta(n)$ children nodes are also created (but not visited), along with their ξ labels.

Following the generation of a structure s, the $\widetilde{\mathcal{B}}$ data structure is updated through back propagation to the root of the tree, ultimately incrementing by $\mathcal{B}(s)$ each value $\widetilde{\mathcal{B}}(v)$ encountered during the generation of s.

Figure 3 illustrates the internal values of $\widetilde{\mathcal{B}}$ after the generation of a pair of structures.

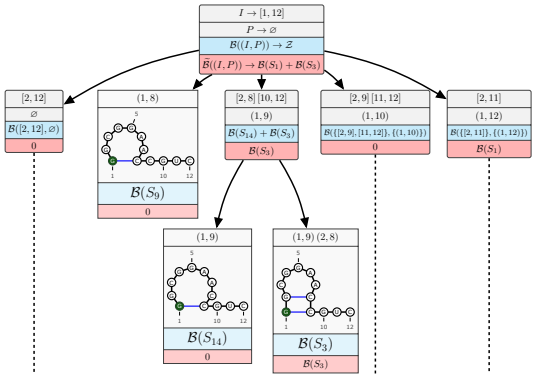


Fig. 3. **Boltzmann factor update** after generating two structures S_1 and S_3 using Algorithm 1. Blue boxes indicate initial partition functions, while red boxes represent the values stored in our dedicated data structure $\widetilde{\mathcal{B}}$ for $\Theta := \{S_1, S_3\}$.

The generation procedure, described in Algorithm 1, can be generalized and optimized to enable a non-redundant generation in the Turner energy model (see Supp. mat. for details).

2.3.3 Complexity analysis

The structure of the modified stochastic backtrack largely follows the one introduced by Ding and Lawrence [11], so a significant overhead to the complexity may only result from the necessity to maintain/update the data structure $\widetilde{\mathcal{B}}$, or from the cost of accessing its stored values $\widetilde{\mathcal{B}}(v)$.

As described above, maintaining the content of a node to reflect a new generation only involves $\mathcal{O}(1)$ operations, except when a new node is visited for the first time, leading to the concrete creation of $\Theta(n)$ children. However, this linear number of operations coincides with the number of alternatives that will be considered by the classic stochastic backtrack. It follows that, in the worst case, the time complexity of the modified backtrack remains only impacted by a constant factor in comparison to the classic backtrack.

Access to $\widetilde{\mathcal{B}}$ can be enabled in $\mathcal{O}(1)$, noting that the (modified) backtrack, at any given step/node v, only needs to access the values of $\widetilde{\mathcal{B}}$ for the current node v and its children. By creating an array of children only when/if needed, and ordering them to match their order of investigation during the backtrack, we ensure access in $\mathcal{O}(1)$ to all the relevant values at every step of the backtrack.

Finally, we note that the cubic precomputation of the partition function remains entirely unmodified. We conclude that the generation of m distinct structures compatible with an RNA of length n can be performed in deterministic $\mathcal{O}(n^3 + mn^2)$ worst-case complexity, matching the complexity of Ding and Lawrence's algorithm [11]. As a final note, this complexity could finally be further improved to $\mathcal{O}(n^3 + mn\sqrt{n})$ by coupling a Boustrophedon backtrack [31] with a *on demand* creation of children nodes (ordering similarly to backtrack) using a dynamic array (only populating the relevant prefix of the children array, with constant overhead).

2.4 Equilibrium statistics from non-redundant samples

Repeated calls to a stochastic backtrack algorithms produce a statistical sample, and are frequently used to estimate statistical properties of RNAs at the thermodynamic equilibrium. While such quantities may sometimes be computed exactly through dynamic programming [8], [9], others seem to induce impractical complexities [10], or are even believed to be associated with NP-hard problems [23]. Sampling-based estimates only require a capacity to evaluate the feature on any given structure, and thus provide a very flexible solution to perform a statistical analysis of thermodynamics ensemble for specific RNAs.

Formally, we define an **RNA feature** as a function $F : \Omega \to \mathbb{R}$, measuring some characteristic of an RNA. Usual features of interest include structural descriptors (presence/absence of base pairs, #helices...), thermodynamic stability (free-energy)... A large number of relevant analysis are then reducible to the **expected value** $\mathbb{E}(F(S))$ of F, where S is a Boltzmann-distributed random structure, defined as

$$\mathbb{E}(F(S)) = \sum_{s \in \Omega} \mathbb{P}(s) \times F(s). \tag{7}$$

Note that this formulation is very general, and encompasses any computable property of RNA structures.

For instance, one may estimate the probability of a motif M occurring within a random – Boltzmann-distributed – random structure as the expected value of a Boolean feature function F_M, taking value $F_M(s) = 1$ if the motif M occurs in s, and 0 otherwise. Indeed, the expectation of F_M then simplifies into the accumulated probability of all structures having M as a motif. More complex statistical quantities, such as the standard deviation of a feature, the Pearson correlation of features, or even higher-order statistics, can be simply obtained by estimating, and combining, powers of the feature(s) of interest.

2.4.1 Empirical mean: The classic redundant (R) estimator

Generally, consider $\mathbf{s} := (s_1, s_2, \ldots, s_m)$ a vector of independent, uniformly-distributed, structures obtained by a redundant (R) sampling. The **empirical mean**, further referred to as the **R estimator**, is given by

$$\widehat{F}(\mathbf{s}) = \frac{1}{m} \sum_{i=1}^{m} F(s_i)$$

This estimator is unbiased, meaning that the expected value of the estimator coincides with the expected value of the random variable F. It is statistically consistent, meaning that its value converges to $\mathbb{E}(F)$ as $m \to \infty$.

However, convergence to the true expectation may be painfully slow, especially in the context of high-variance features, couples with RNAs where the best part of the probability mass accumulates over a few structures. For instance, we describe in Figure 1 the frequencies of occurrences obtained from 50 independent generations. The probability of the nucleotide at position 1 being paired at the thermodynamic equilibrium can be estimated as the expectation of a feature F_1 taking value 1 if nucleotide 1 is paired (*e.g.* structure S_1), or 0 otherwise (*e.g.* structure S_2). The sample illustrated by Figure 1 induces an empirical mean of $\widehat{F_1} = 0.78$, which differs substantially from the real probability of 0.87.

2.4.2 Non-redundant samples require custom estimators

Intuitively, in the context of independently-generated elements, the empirical mean (R estimator) can be seen as solving two tasks simultaneously:

- Estimating the probability of structures;
- Computing a weighted average of the feature values.

However, within a Boltzmann distribution, the exact probability of a structure is entirely determined by its free-energy, and available (or typically computable with a $\Theta(n)$ time postprocessing) as soon as the structure is produced. Redundancy is then, in theory, uninformative and could be avoided.

Unfortunately, one cannot simply use the empirical mean while processing a non-redundant sampled set of structures. Indeed, the naive estimator implicitly assigns the same weight to all structures found in the sample, and redundancy is then crucial to account for the Boltzmann probabilities of structures. Moreover, natural alternatives, such as weighted averages based on the probability/Boltzmann factor, represent biased estimators. To further illustrate such

a bias, consider the following *common sense* weighted estimator:

$$\bar{F}(\mathbf{s}) = \sum_{i=1}^{m} \frac{\mathbb{P}(s_i)}{\sum_j \mathbb{P}(s_j)} \times F(s_i).$$

Now let us consider a trivial feature $F^\star$, defined from our example as $F^\star(S) := \{1 \text{ if } S = S^\star; \text{ or } 0 \text{ otherwise}\}$ where $S^\star$ is a fixed structure of interest, such that one has $\mathbb{E}(F^\star) = \mathbb{P}(S^\star)$. For a sample consisting of a unique structure, $\bar{F}$ coincides with $\widehat{F}$, and the estimator is therefore unbiased. However, as soon as two structures (s', s'') are generated in this order in a non-redundant manner ($s' \neq s''$), three cases may arise:

1) $S^\star$ is emitted first: $s' = S^\star \implies s'' \neq S^\star$;
2) $S^\star$ is emitted in second: $s'' = S^\star \implies s' \neq S^\star$;
3) None of the two structures is $S^\star$: $S^\star \notin \{s', s''\}$.

The third case does not contribute to the expectation of the weighted estimator, so we get

$$\begin{aligned}\mathbb{E}(\bar{F}^\star) &= \sum_{s',s''} \mathbb{P}((s',s'')) \times \bar{F}^\star((s',s'')) \\ &= \sum \begin{cases} \sum_{s'':=s\neq S^\star} \frac{\mathbb{P}(S^\star).\mathbb{P}(s)}{1-\mathbb{P}(S^\star)} \times \frac{\mathbb{P}(S^\star)}{\mathbb{P}(S^\star)+\mathbb{P}(s)} & \{s' = S^\star\} \\ \sum_{s':=s\neq S^\star} \frac{\mathbb{P}(s).\mathbb{P}(S^\star)}{1-\mathbb{P}(s)} \times \frac{\mathbb{P}(S^\star)}{\mathbb{P}(s)+\mathbb{P}(S^\star)} & \{s'' = S^\star\}. \end{cases}\end{aligned}$$

The value of $\mathbb{E}(\bar{F}^\star)$ generally differs from $\mathbb{P}(S^\star)$, as hinted by its quadratic dependency on $\mathbb{P}(S^\star)$.

For instance, consider the distribution such that $\mathbb{P}(S^\star) = 1/2$ and $\mathbb{P}(s) := 1/\kappa, \forall s \neq S^\star$ with $\kappa := 2(|\Omega| - 1) > 2$. The above expression simplifies, and we get

$$\mathbb{E}(\bar{F}^\star) = \frac{\kappa^2}{2(\kappa-1)(\kappa+2)} < \frac{1}{2}.$$

It follows that, whenever Ω has more than 2 structures, the empirical estimator is not guaranteed to be unbiased for a non-redundant sample.

2.4.3 Estimating from a non-redundant (NR) sample

We now introduce our novel **non-redundant (NR) estimator** $\widetilde{F}(\mathbf{t})$ for the expected value of a feature F from a non-redundant sample generated according to the distribution 2. Given a non-redundant sequence of sampled structures $\mathbf{t} := (t_1, t_2, \ldots, t_m)$, it is defined as:

$$\widetilde{F}(\mathbf{t}) = \frac{1}{m} \sum_{i=1}^{m} F(t_i) \left(1 - \bar{F}^{(0)}_{\Theta_{i-1}} + (m-i) \times \mathbb{P}(t_i)\right) \quad (8)$$

where $\Theta_i := (t_1, \ldots, t_i)$ and $\bar{F}^{(x)}_{\Theta} := \sum_{t\in\Theta} \mathbb{P}(t)\, F(t)^x$.

Intuitively, the estimator can be seen as a weighted sum, augmented with correction terms to compensate: i) the inflated probability of sample not generated at a given point; ii) the absence of previously-generated structures from future iterations. Namely, the $1 - \bar{F}^{(0)}_{\Theta}$ term can be interpreted as correcting for the fact that, at the i-th iteration of the non-redundant sampling algorithm, an overall probability mass of $\bar{F}^{(0)}_{\Theta}$ has already been generated, increasing the probability of generating t_i by a factor $1/(1 - \bar{F}^{(0)}_{\Theta})$. Similarly, the term $(m-i)\times\mathbb{P}(t)$ coincides with the expected number of futures occurrences for t_i using classic sampling, correcting for its absence in the subsequent elements of the non-redundant sample.

Note that, as long as the accumulated probability $\bar{F}^{(0)}_{\Theta}$ of previously-generated structures remains negligible, the product $(m-i) \times \mathbb{P}(t_i)$ remains close to zero. The NR estimator is then equivalent to the empirical mean, consistent with the fact that redundant sampling then typically yields a non-redundant sequence of structures.

Computing the NR estimator typically induces **negligible time and space overhead**, in comparison to the sampling itself, assuming that the feature F of interest can be evaluated in time $\mathcal{O}(n)$. Namely, the sum in Equation (8) can then be computed in time $\Theta(m \times n)$, since:

1) Individual probabilities $\mathbb{P}(t_i)$ can be computed in constant time;
2) The sums involved in the computation of $\bar{F}^{(0)}_{\Theta}$ can be incrementally updated in constant time anytime a new value becomes available, and so can $\widetilde{F}(\mathbf{t})$.

The overall complexity of computing the estimator compares favorably against the $\Theta(n^3 + m\,n^2)$ time required by the (non-redundant) sampling itself.

2.4.4 Properties of non-redundant estimator

Firstly, the NR estimator is **unbiased**, meaning that the expected value of the estimator over a sequence of random structures. More precisely, we establish in Supp. mat. that

$$\mathbb{E}\left(\widetilde{F}(\mathbf{t})\right) = \mathbb{E}\left(F(S)\right)$$

where S is a random Boltzmann-distributed structure, and $\mathbf{t}$ a non-redundant sequence of random structures consisting of at least one structure. This follows the estimator admitting an alternative – equivalent – form as

$$\widetilde{F}(\mathbf{t}) = \frac{1}{m}\left(\sum_{i=1}^{m} F(v_i) \times (1 - \bar{F}^{(0)}_{\Theta_{i-1}}) + \bar{F}^{(1)}_{\Theta_{i-1}}\right). \quad (9)$$

Then, it is possible to prove (see Supp. Mat. for details) that, for any set Θ of avoided structures, one has

$$\mathbb{E}\left(F(T) \times (1 - \bar{F}^{(0)}_{\Theta}) + \bar{F}^{(1)}_{\Theta} \mid \Theta\right) = \mathbb{E}\left(F(S)\right).$$

The NR estimator can then be reformulated as the average over a sequence of random variables, each having $\mathbb{E}\left(F(S)\right)$ as their expected value. Since the expectation of a sum equals the sum of expectations, the unbiased nature of the estimator immediately follows.

The absence of bias implies that the NR estimator can also be used to **estimate the variance** of a feature $F(T)$. To that end, simply compute the NR estimator for the features F and F^2, to obtain estimates for the expectations of $F(S)$ and $F(S)^2$ respectively, and finally use the formula

$$\mathbb{V}\left(F(S)\right) = \mathbb{E}\left(F(S)^2\right) - \mathbb{E}\left(F(S)\right)^2$$

to recover an estimate for the variance.

Secondly, the NR estimator is **statistically consistent**, *i.e.* as the number of sample grows, the estimated value

gets increasingly and arbitrarily close to the real expected value of the feature. Since this property only formally holds for infinite sources, we consider a generalized version of the NR sampling process, which repeatedly returns a fake structure $\perp$ with probability 0 and value $F(\perp) = 0$ once the full collection of structure has been generated. Remark that, for any i-th sample such that $i \geq |\Omega|$ samples, the contribution to the sum in the estimator greatly simplifies, since $1 - \bar{F}^{(0)}_{\Theta_{i-1}} = 0$ and $\bar{F}^{(1)}_{\Theta_{i-1}} = \mathbb{E}(F(S))$. It follows that, denoting by A_Ω the accumulated value of the sum over the $|\Omega|$ first samples, one has

$$\widetilde{F}(\mathbf{t}) = \frac{1}{m}\left(A_\Omega + \sum_{i=|\Omega|+1}^{m} \mathbb{E}(F(S))\right) = \frac{A_\Omega}{m} + \frac{(m-|\Omega|)}{m} \cdot \mathbb{E}(F(S)).$$

It immediately follows that

$$\lim_{m\to\infty} \widetilde{F}(\mathbf{t}) = \mathbb{E}(F(S)),$$

implying the consistency of the estimator.

Finally, the NR estimator provably has **lower variance** than the empirical mean computed from a redundant sample, using the same number of structures. Formally, one has

$$\mathbb{V}(\widetilde{F}(\mathbf{t})) \leq \mathbb{V}(\widehat{F}(\mathbf{s})), \forall |\mathbf{t}| = |\mathbf{s}| \geq 1,$$

and the inequality is even provably strict when $|\mathbf{t}| = |\mathbf{s}| > 1$. It implies a lower dispersion for the values obtained using the NR estimator and, generally, a faster convergence to the exact value. A formal proof of this property is technically involved, and can be found in Supplementary Material. It is worth noting that this theoretical superiority has concrete practical consequences, as can be observed in Figure 4.

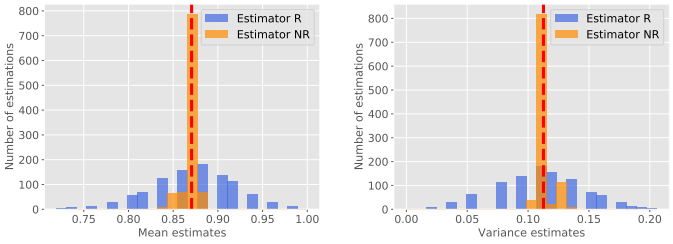


Fig. 4. **Non-redundant sampling/estimators (NR) provides more accurate estimates than traditional sampling (R)/empirical mean.** For GGCGGAACCGUC, we consider the Boolean feature indicating the base pairing status of the first nucleotide (unpaired $\rightarrow$ 0 , paired $\rightarrow$ 1). We estimate the feature expectation (*i.e.* probability of being paired; left) and variance (right) from 50 secondary structures. Dashed lines indicate the exact value of both statistics.

3 Results

Our non-redundant backtracking procedure was implemented within the Vienna RNA package [12], supporting recent versions of the Turner energy model. NR sampling is available within the `RNAsubopt`, `RNApvmin` and `RNAalifold` utilities using the `-N` modifier. The Vienna RNA package can be compiled from freely-accessible sources, or downloaded as a bundle of binaries for virtually any architectures at:

https://www.tbi.univie.ac.at/RNA/

Experiments were implemented in `python`. Implementations for NR generation (Simulation using `ViennaRNA`, NR estimator, ...) and a tutorial are freely-accessible sources at:

https://gitlab.com/christelle.rovetta/rnanr-stats

3.1 Inferring dot-plots

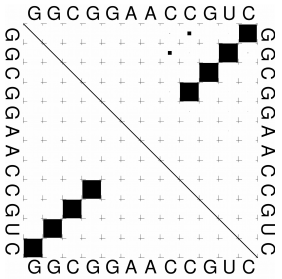


Fig. 5. **Exemplary RNA dot plot.** A dot plot diagram allows the visualization of the most probable (Min. Free-Energy) structure (lower-left triangular region; black squares indicating presence/absence of base pairs), jointly with the base pairing probability matrix (upper-right triangular region; area of black squares being proportional to probability).

As a first illustration of the potential of our NR methodology, we showcase the fast computation of estimates for the **base pair probability matrices**, sometimes referred to as **dot plots**. Such matrices are at the core of reference computational methods in RNA bioinformatics, including structural alignment [32] and design [33]. Dot plots contain the probabilities $p_{i,j}$ of forming a base pair between positions i and j at the thermodynamic equilibrium, such that

$$p_{i,j} = \sum_{\substack{s\in\Omega,\\(i,j)\in s}} \mathbb{P}(s).$$

Figure 5 illustrates the dot-plot (upper-right triangle) and MFE structure (lower-left; Turner model) for our running example GGCGGAACCGUC.

Exact probabilities can be computed using a variant of the inside/outside algorithm, practically doubling the time of computing the partition function [3]. As an alternative, we consider the estimation of such probabilities based on Boolean features $D_{i,j}$ that indicate presence/absence of a base pair (i,j):

$$D_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in s \\ 0 & \text{otherwise.} \end{cases}$$

The expectation of any such feature is given by

$$\mathbb{E}(D_{i,j}) = \sum_{s\in\Omega} D_{i,j} \times \mathbb{P}(s) = \sum_{\substack{s\in\Omega,\\(i,j)\in s}} 1 \times \mathbb{P}(s) = p_{i,j}.$$

An estimator for the expectation of the proposed feature is therefore also an estimator for the base-pair probabilities.

We consider the dot plots estimates $\{\widehat{D}_{i,j}(\mathbf{s})\}_{i,j}$ and $\{\widetilde{D}_{i,j}(\mathbf{t})\}_{i,j}$, obtained using the empirical mean and our NR estimator respectively. We assess the accuracy of both estimators by comparing their inferred probabilities to the exact base pair probabilities, computed using the `ViennaRNA`

implementation of the McCaskill algorithm [3]. The **overall error** e_{R} and e_{NR}, respectively induced by the empirical mean and the NR estimator, are defined as

$$e_{\mathrm{R}} = \frac{\sqrt{\sum_{i,j}(\widehat{D}_{i,j} - p_{i,j})^2}}{n(n-1)} \text{ and } e_{\mathrm{NR}} = \frac{\sqrt{\sum_{i,j}(\widetilde{D}_{i,j} - p_{i,j})^2}}{n(n-1)}.$$

The renormalization by $n(n-1)$ is used to mitigate the influence of the sequence length, and thus assess the average error per base pair probability.

Next, we turn to the analysis of both estimators in two settings: First, we compare the accuracy of both estimates, computed from two sets of structures having equal cardinality; Then, to account for the fact that non-redundant sampling typically induces a 10 to 20% computational overhead [34], we allocate the same time to both generators, and compute estimates for a given elapsed time.

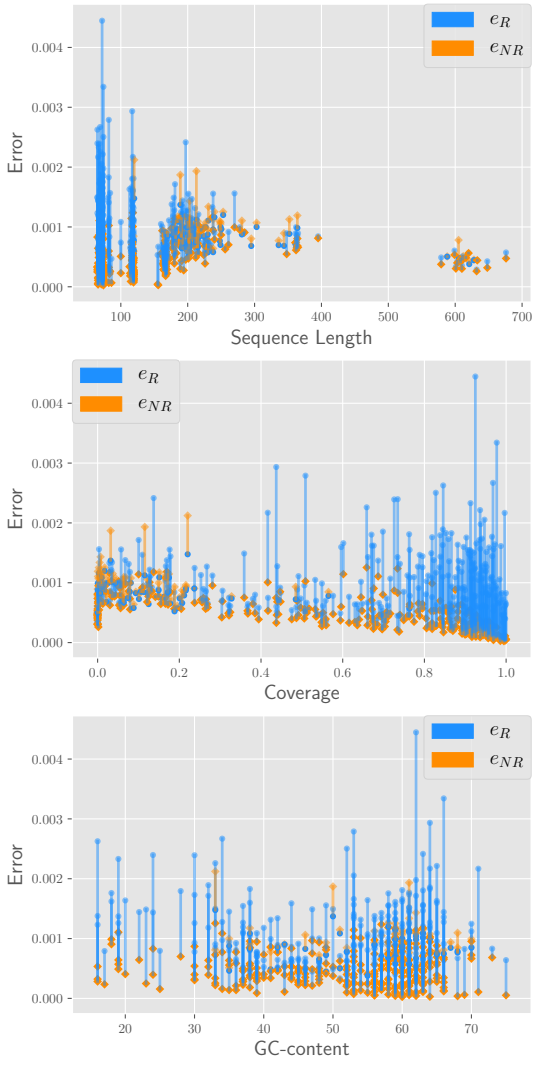


Fig. 6. **Effect of sequence length, coverage and GC% on the accuracy of redundant and non-redundant dot-plot estimators.** Comparison of errors e_{R} (redundant estimator $\widehat{D}$, in blue) and e_{NR} (non-redundant estimator $\widetilde{D}$, in orange), for $m = 1000$. To emphasize the difference between the two estimators for a single sequence, a line is drawn between the difference between e_{R} and e_{NR} is shown in color of more important error. The sequences are ordered by their length (top), coverage (middle) and %GC (bottom)

3.1.1 *For a fixed sample size*

Here we consider the errors observed by analyzing a sample of fixed cardinality $m = 1000$, generated using R and NR sampling. We gathered a data set of 365 sequences, extracted from the *seed* alignments of selected RFAM [1] families, chosen to cover a wide range of lengths and GC-contents:

- 63 sequences from RF00001 – lengths 91 to 135 nts;
- 100 sequences from RF00005 – lengths 62 to 93 nts;
- 60 sequences from RF00061 – lengths 177 to 365 nts;
- 72 sequences from RF00174 – lengths 168 to 248 nts;
- 20 sequences from RF01071 – lengths 395 to 676 nts;
- 50 sequences from RF01731 – lengths 66 to 173 nts.

We evaluated both estimators with respect to the above metrics, sampling $m = 1000$ secondary structures both redundantly and non-redundantly for each of the 365 sequences, and computing $\widehat{D}$ and $\widetilde{D}$. We also computed the ground truth D using the RNAfold implementation of the McCaskill algorithm [3], from which we derived the values of e_{R} and e_{NR}.

The results shown in Figure 6 reveal that, for samples of equal cardinality, the error e_{R} of classic (redundant) sampling/estimator is overwhelmingly larger than the error e_{NR} induced by our non-redundant sampling/estimator (*i.e.* the difference $e_{\mathrm{R}} - e_{\mathrm{NR}}$ is rarely negative). More specifically, the non-redundant estimate $\widetilde{D}$ achieves a lower error than its redundant counterpart $\widehat{D}$ for 83.3% of the sequences, as shown by the histogram in Figure 7. Non-redundant estimates are substantially better for higher values of coverage, and shorter sequences, the NR estimator being particularly relevant for sequences that are shorter than 200 nt nucleotides. Interestingly, we did not observe a substantial impact of the GC-content on the accuracy of both estimators.

This demonstrates that non-redundant sampling, in combination with a non-redundant estimator, can provide more accurate estimates than using the empirical mean based on a classic redundant sample.

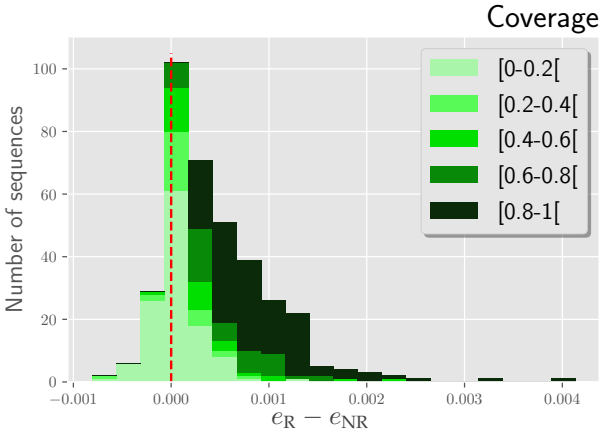


Fig. 7. **Histogram of the difference in overall errors between R and NR estimators for dot-plot given a fixed sample size.** Samples of a fixed cardinality $m = 1000$ are provided to both estimators for all sequences in our RFAM-based dataset. Bins are broken up by coverage, confirming that NR sampling is increasingly dominant for more concentrated distributions.

3.1.2 For a prescribed runtime

In practice, producing a NR sample requires slightly more time than a redundant sample of the same cardinality. This computational overhead, which only represents a fraction of the original running time of the sampling phase, is due to the additional operations involved in maintaining the dedicated data structure, and accessing its values during the sampling. For this reason, it seems more fair to compare the performances on both estimators when allocating the same amount of time to both sampling procedures.

In order to assess the evolution of error as a function of the elapsed time, we performed a detailed analysis of a subset of sequences, selected to cover a wide range of sequence length. The reduced data set includes sequences:

- **a** `X06837.1/1-119` (100nt – RF00001);
- **b** `M30199.1/68-167` (119nt – RF00001);
- **c** `BAAU01027214.1/624-783` (160nt – RF01731).
- **d** `CP000283.1/2593935-2594143` (209nt–RF00174);
- **e** `AY344021.1/1-348` (348nt – RF00061);
- **f** `CP000679.1/1996671-1997302` (632nt – RF01071).

We first sampled a nominal number of unique structures using NR sampling, storing the final elapsed time $t^\star$. We then computed the NR estimator for each subset of structure generated after time t representing fractions of $t^\star$, with $t/t^\star \in [0, 1/8, 1/4, 1/3, 1/2, 2/3, 3/4, 7/8, 1]$. Finally, we computed the error for a redundant set of structures generated using the same time.

As can be seen in Figure 8, the NR estimator clearly outperforms its competitor in 4 out of the 6 cases, and essentially matches its competitor for one the remaining two. Surprisingly, the NR estimator yields a smaller error e_{NR} for `CP000679.1/1996671-1997302`, the longest among the sequences presented here. This behavior may be attributed to the natural stochasticity of statistical estimators, noting that the NR estimator gently degrades into the empirical mean when the sampled structure only represent a negligible proportion of the Boltzmann probability distribution. By contrast, the dominance of the NR estimator over the empirical mean is much clearer, and robust, within smaller sequences, where redundancy has a much greater chance to manifest itself.

3.2 Sampling distinct shapes and estimating shape probabilities

RNA Shapes are abstract representations of secondary structures. Initially introduced as tool for the comparative folding of RNA, developed by group of Robert Giegerich within a body of work spanning more than a decade [23], [35]. RNA shapes represent coarse-grained versions of the classic secondary structures, focusing on the high-level organization of RNA architectures. As such, RNA Shapes are less sensitive to minor perturbations, such as insertions and deletions of individual nucleotides, than the classic secondary structure. They can thus be used to extract recurrent conformations across homologous RNAs without having to align their sequence.

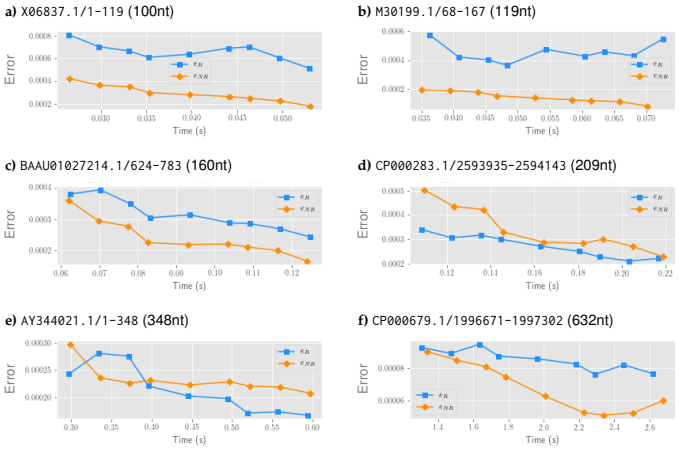


Fig. 8. **Evolution with time of the prediction error of base-pair estimators.** For 6 sequence spanning a large range of lengths, we report the errors of the redundant ($\widehat{F}$ – blue) and non-redundant ($\widetilde{F}$ – orange) estimators.

At its coarsest level, the **RNA Shapes** associated with a structure is obtained by suppressing all unpaired positions, and contracting consecutive (*aka* stacking) pairs into a single pair of matching brackets. Shapes can be represented using a notation that is analogous to the classic dot-parenthesis notation for secondary structures, using brackets instead of parentheses.

For instance, the structure S_1 in Figure 1, which consists of the set of base pairs $\{(1, 12), (2, 11), (3, 10)\}$ admits a representation `(((.....)))` in dot-parenthesis notation, and the shape associated with S_1 is simply denoted by

$$\textsc{Shape}(S_1) = \texttt{[]}.$$

For a more complex example, the secondary structure $S^\star$ = `(((..))..(((...)))))` admits $\textsc{Shape}(S^\star)$ = `[[][]]` as its coarsest shape representation.

Notably, a single shape typically represent a large number of, structurally similar, secondary structures. Computing the overall Boltzmann probability of a shape can be done in polynomial time [36], but it requires a complex and shape-specific computation. Moreover, no deterministic efficient algorithm is currently known for computing the list of RNA shapes ordered by decreasing Boltzmann probability, and current methods typically resort to (redundant) statistical sampling to identify promising shape candidates [23]. This suggests two contexts in which non-redundant sampling and estimator could be beneficial: i) The computation of a list of dominant shapes; and ii) The estimation of the probability of a given shape.

3.2.1 Comprehensive lists of dominant shapes

As a straightforward application, one can use NR sampling in order to establish a more comprehensive list of shapes supported by an RNA sequence, *i.e.* shapes π such that $\textsc{Shape}(S) = \pi$ for some secondary structure S. In this context, redundancy is clearly uninformative, and using NR sampling represents a natural choice.

Indeed, as can be seen in Figure 9, using NR sampling offers sizable gains, allowing to access a more comprehen-

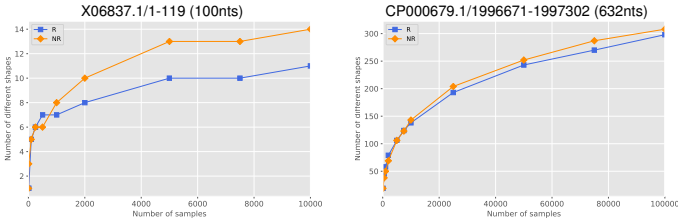


Fig. 9. **Number of different RNA shapes** populated by at least one secondary structure within samples of increasing cardinality, produced using R (blue) and NR sampling (orange).

sive list of shapes for smaller RNAs. This benefit decreases for longer RNAs, for which redundancy seldom occurs, although some gain can still be observed. Nevertheless, even for longer RNAs, the non-linear behavior of both curves suggests a high level of redundancy at the shape level, while the superposition of curves indicates an absence of redundancy at the secondary structure level.

3.2.2 *Estimating the probability of shapes*

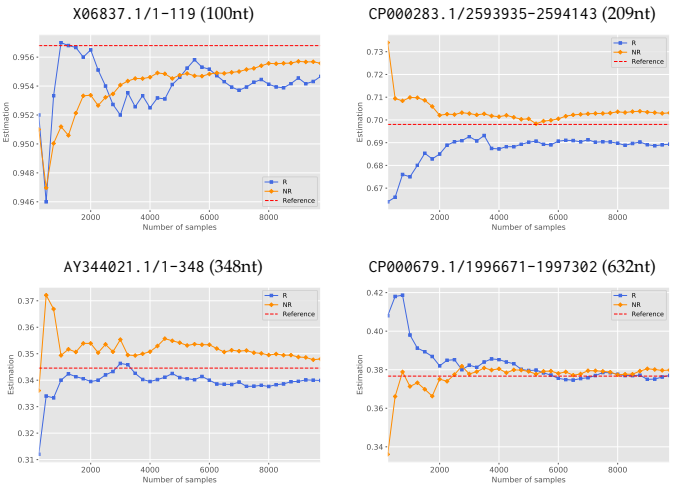


Fig. 10. **Estimation of MFE shape probability** for sample size m ranging from 250 to 10 000. Empirical mean (R) estimates $\widehat{F}$ drawn in blue, with non-redundant estimates $\widetilde{F}$ in orange. Reference value colored in red.

In this second application, we use NR sampling to estimate the Boltzmann probability associated with a given shape. While any shape can theoretically be considered, we choose to focus on **the shape associated with the minimum free-energy (MFE)** secondary structure, *i.e.* the most stable structure. This structure is also the most probable at the thermodynamic equilibrium.

To achieve this objective, we define a feature function $F_{\text{SHAPE}} : \Omega \to \mathbb{B}$ such that

$$F_{\text{SHAPE}}(S) = \begin{cases} 1 & \text{if } \text{SHAPE}(S) = \text{SHAPE}(\text{MFE}) \\ 0 & \text{otherwise.} \end{cases}$$

Again, it can be shown that the expected value of F_{SHAPE} coincides with the probability of generating a structure admitting the MFE shape as its representative.

To compare the quality of estimates, we consider sequences **a**, **d**, **e**, and **f** from our reduced dataset, and report the evolution of the R/NR estimates for the expectation of F_{SHAPE}, *a.k.a.* the MFE shape probability, for sample sizes $m \in [250, 500, \ldots, 10000]$. We used the empirical estimate computed for $\widehat{F}$ for m =1 000 000 samples as reference/ground truth.

The results, which can be visualized in Figure 10, reveal once again that NR estimates typically outperform the classic empirical mean. In particular, NR estimate tend to show a smoother convergence towards the reference value, as shown in Figure 11, as well as a better concentration around the reference for a given sample size.

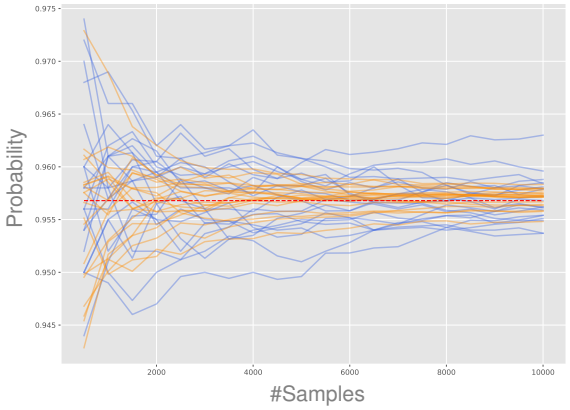


Fig. 11. **Concentrations of R (blue) and NR (orange) estimates of shape probabilities.** For sequence `X06837.1/1-119` (100nt), 20 independent sampling of m =10 000 structures are performed using both generator, and the evolution of estimates with an increasing sample size is plotted. A red line indicates the reference probability.

4 On the Right Number of Structures

When using sampling to estimate statistical properties of RNA structures at the thermodynamic equilibrium, a recurrent – crucial – question is to choose the number of generated samples as to produce accurate estimates. Historically, and in many subsequent works, a sample size of 1 000 structures has been proposed [11], somewhat irrespectively of the context. However, such a *one size fits all* may not yield accurate, or reasonably reproducible results, motivating the probabilistic analysis below.

Before stating our recommendations, we need to remind the crucial concept of **confidence interval**. In general, a sample size needs to be chosen in order to achieve a desired level of precision. However, since the process of sampling is stochastic in nature, it is impossible to unconditionally guarantee a given precision since, out of the possible sequences of generated structures, some may typically induce arbitrarily large errors. For instance, for `GGCGGAACCGUC` (*c.f.* Figure 1), redundant sampling may generate structure S_{84} m times, leading to an (erroneous) estimated probability of 1 for base pairing the first nucleotide. However, this scenario has an abysmal probability, lower than 10^{-9m}, so one needs to adopt a **confidence intervals** perspective, considering the trade-off between the precision and how often this precision is achieved while estimating from a random sample.

In the case of the R estimator, the empirical mean is essentially a sum of independent variables, meaning that

Tolerated Error	Frequency within tolerance 90%	95%	99%
$\varepsilon = 20\%$	37	46	66
$\varepsilon = 10\%$	150	184	265
$\varepsilon = 5\%$	599	738	1 060
$\varepsilon = 2.5\%$	2 397	2 951	4 239
$\varepsilon = 1\%$	14 979	18 444	26 492
$\varepsilon = 5‰$	59 915	73 778	105 966
$\varepsilon = 1‰$	1 497 866	1 844 440	2 649 159

TABLE 2
Recommended number of samples for estimating equilibrium probabilities (boolean features). For instance, to ensure that the estimate falls within 1% of the true value for 95% of the runs, a large number of m =18 444 structures should be generated.

classic concentration inequalities contributed by the field of probability theory, can be used with minimal modifications. In particular, the **Hoeffding inequality** implies that, for any feature F:

$$\mathbb{P}\left(|\widehat{F}(\mathbf{S}) - \mathbb{E}\left(F(S)\right)| \geq \varepsilon\right) \leq 2\exp\left(\frac{-2m\varepsilon^2}{c}\right), \qquad (10)$$

where ε is a tolerated level of error, $\mathbf{S}$ is a random sample of size m and $c := (\max_S(F(S)) - \min_S(F(S)))^2$. Note that when a feature function only takes values 0 or 1, as in many of our experiments, then one has $c = 1$. Equation (10) can be used to build a **confidence interval** at level $(1-\alpha)$, for any value $\alpha \in [0,1]$:

$$\left[\widehat{F}(\mathbf{S}) - \sqrt{\frac{c}{2m}\log\left(\frac{2}{\alpha}\right)}, \widehat{F}(\mathbf{S}) + \sqrt{\frac{c}{2m}\log\left(\frac{2}{\alpha}\right)}\right].$$

This means that, over multiple executions of the R sampling/estimation, at least a fraction $(1-\alpha)$ of the runs will feature an error smaller than $\sqrt{\frac{c}{2m}\log\left(\frac{2}{\alpha}\right)}$. This function can be inverted numerically to estimate the number m of samples that achieve an error bounded by ε at least $(1-\alpha)$ of the times.

We report in Figure 2 some typical sample sizes required to achieve a given precision with reasonable probability when estimating probabilities (*i.e.* expectations of 0/1-valued features). For instance, to reach a **90% chance** of estimating a base pair probability **within 0.5% of its true value**, a total of **59 915 structures** should be generated.

By contrast, the **1 000 structures** usually considered in the literature will guarantee a value **within 3%** of the true probability **only 2/3 of the times**, although this sample size will **almost always** (99%) return estimates **within 5%** of the correct value. Finally, the formula can be adapated to more complex features, taking values in a wider range. For instance, to compute the **expected distance** for sequence **a** (100 nts $\rightarrow c = 99$), a **sample of 263 structures** will produce an estimated distance **within one step from the true value** in more than **99% of executions**.

Due to its lower variance, our NR sampling/estimator achieves strictly better estimates for a given sample size, leading to more modest requirements. However, a refined analysis would be much more challenging due to the dependence of consecutive samples. Therefore, we recommend sampling the same number of structures as described above for the R sampling, but expect more accurate results.

5 Conclusion

In this work, we have described an algorithm for the non-redundant sampling of secondary structures in the Boltzmann ensemble, using algorithmic principles introduced by some of the authors [15], [16]. This algorithm was implemented in the Vienna RNA package [12], and is currently available as an extension of the `RNAsubopt` tool. While the non-redundant sampling allows to produce more diverse samples, it induces dependencies between structures generated during the sampling, forbidding the use of classic estimators, such as the empirical mean. We have thus introduced a statistical estimator for non-redundant sampling which we proved is both unbiased, consistent, easily computed. By exploiting explicit knowledge of the emission probability of structures within the Boltzmann distribution, we showed that our new estimator produces higher-quality estimates (lower variance) than classic estimators based on redundant samples of the same cardinality. Empirically, we demonstrated that our non-redundant sampler and estimator achieves better estimates for various quantities of interest at the thermodynamic equilibrium. We concluded this study with recommendations regarding the sample size to achieve reproducibility.

While this work describes specific applications of non-redundant sampling to RNA bioinformatics, its scope of application is much wider. Boltzmann-Gibbs distributions are quite frequent in Bioinformatics [37], [22], [38], and could be explored to study the stability of predictions in any context where unambiguous dynamic-programming schemes [9], [39] exist. Our novel estimator can then be used without any modifications, for instance to estimate the diversity of near-optimal solutions.

Acknowledgments

The authors are greatly indebted to Ivo Hofacker for earlier discussions, motivating the search and discovery of a specific estimator from non-redundant samples. This work was part of the `RNALands` project, jointly supported by the French *Agence Nationale de la Recherche* (ANR-14-CE34-0011) and the Austrian *Fonds zur Förderung der wissenschaftlichen Forschung* (I 1804-N28).

References

[1] I. Kalvari, E. P. Nawrocki, J. Argasinska, N. Quinones-Olvera, R. D. Finn, A. Bateman, and A. I. Petrov, "Non-coding RNA analysis using the RFAM database." *Current protocols in bioinformatics*, vol. 62, p. e51, Jun. 2018.

[2] L.-L. Zheng, J.-H. Li, J. Wu, W.-J. Sun, S. Liu, Z.-L. Wang, H. Zhou, J.-H. Yang, and L.-H. Qu, "deepBase v2. 0: identification, expression, evolution and function of small RNAs, LncRNAs and circular RNAs from deep-sequencing data," *Nucleic acids research*, vol. 44, no. D1, pp. D196–D202, 2015.

[3] J. S. McCaskill, "The equilibrium partition function and base pair binding probabilities for RNA secondary structure," *Biopolymers: Original Research on Biomolecules*, vol. 29, no. 6-7, pp. 1105–1119, 1990.

[4] P. Clote, Y. Ponty, and J.-M. Steyaert, "Expected distance between terminal nucleotides of RNA secondary structures." *Journal of mathematical biology*, vol. 65, pp. 581–599, Sep. 2012.

[5] E. Freyhult, V. Moulton, and P. Clote, "RNAbor: a web server for RNA structural neighbors." *Nucleic acids research*, vol. 35, pp. W305–W309, Jul. 2007.
[6] R. Lorenz, C. Flamm, and I. L. Hofacker, "2d projections of RNA folding landscapes," in *German conference on bioinformatics 2009*. Gesellschaft für Informatik eV, 2009.
[7] J. Waldispühl, S. Devadas, B. Berger, and P. Clote, "RNAmutants: a web server to explore the mutational landscape of RNA secondary structures." *Nucleic acids research*, vol. 37, pp. W281–W286, Jul. 2009.
[8] I. Miklós, I. M. Meyer, and B. Nagy, "Moments of the boltzmann distribution for RNA secondary structures." *Bulletin of mathematical biology*, vol. 67, pp. 1031–1047, Sep. 2005.
[9] Y. Ponty and C. Saule, "A combinatorial framework for designing (pseudoknotted) RNA algorithms," in *Algorithms in Bioinformatics*, ser. LNBI, M.-F. S. T. Przytycka, Ed. Springer, Jan. 2011, no. 6833, pp. 250–269. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-23038-7_22
[10] J. Qin, M. Fricke, M. Marz, P. F. Stadler, and R. Backofen, "Graph-distance distribution of the boltzmann ensemble of RNA secondary structures." *Algorithms for molecular biology : AMB*, vol. 9, p. 19, 2014.
[11] Y. Ding and C. E. Lawrence, "A statistical sampling algorithm for RNA secondary structure prediction." *Nucleic acids research*, vol. 31, pp. 7280–7301, Dec. 2003.
[12] R. Lorenz, S. H. Bernhart, C. Höner Zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, and I. L. Hofacker, "ViennaRNA package 2.0." *Algorithms for molecular biology : AMB*, vol. 6, p. 26, Nov. 2011.
[13] J. S. Reuter and D. H. Mathews, "RNAstructure: software for RNA secondary structure prediction and analysis." *BMC bioinformatics*, vol. 11, p. 129, Mar. 2010.
[14] N. R. Markham and M. Zuker, "UNAFold: software for nucleic acid folding and hybridization." *Methods in molecular biology (Clifton, N.J.)*, vol. 453, pp. 3–31, 2008.
[15] W. A. Lorenz and P. Clote, "Computing the partition function for kinetically trapped RNA secondary structures." *PloS one*, vol. 6, p. e16178, Jan. 2011.
[16] J. Michálik, H. Touzet, and Y. Ponty, "Efficient approximations of RNA kinetics landscape using non-redundant sampling." *Bioinformatics (Oxford, England)*, vol. 33, pp. i283–i292, Jul. 2017.
[17] J. Waldispühl and Y. Ponty, "An unbiased adaptive sampling algorithm for the exploration of RNA mutational landscapes under evolutionary pressure." *Journal of computational biology : a journal of computational molecular cell biology*, vol. 18, pp. 1465–1479, 2011.
[18] M. Kucharík, I. L. Hofacker, P. F. Stadler, and J. Qin, "Basin hopping graph: a computational framework to characterize RNA folding landscapes." *Bioinformatics (Oxford, England)*, vol. 30, pp. 2009–2017, Jul. 2014.
[19] S. Pei, J. S. Anthony, and M. M. Meyer, "Sampled ensemble neutrality as a feature to classify potential structured RNAs." *BMC genomics*, vol. 16, p. 35, Feb. 2015.
[20] A. Spasic, S. M. Assmann, P. C. Bevilacqua, and D. H. Mathews, "Modeling RNA secondary structure folding ensembles using SHAPE mapping data." *Nucleic acids research*, vol. 46, pp. 314–323, Jan. 2018.
[21] S. Washietl, I. L. Hofacker, P. F. Stadler, and M. Kellis, "Rna folding with soft constraints: reconciliation of probing data and thermodynamic secondary structure prediction." *Nucleic acids research*, vol. 40, pp. 4261–4272, May 2012.
[22] V. Reinharz, Y. Ponty, and J. Waldispühl, "A weighted sampling algorithm for the design of RNA sequences with targeted secondary structure and nucleotide distribution." *Bioinformatics (Oxford, England)*, vol. 29, pp. i308–i315, Jul. 2013.
[23] B. Voss, R. Giegerich, and M. Rehmsmeier, "Complete probabilistic analysis of RNA shapes." *BMC biology*, vol. 4, p. 5, Feb. 2006.
[24] D. Gardy and Y. Ponty, "Weighted random generation of context-free languages: Analysis of collisions in random urn occupancy models," in *GASCOM - 8th conference on random generation of combinatorial structures - 2010*. Montréal, Canada: LACIM, UQAM, Sep. 2010, p. 14pp. [Online]. Available: https://hal.inria.fr/inria-00543150
[25] J. Du Boisberranger, D. Gardy, and Y. Ponty, "The weighted words collector," in *AOFA - 23rd International Meeting on Probabilistic, Combinatorial and Asymptotic Methods for the Analysis of Algorithms - 2012*, ser. DMTCS Proceedings, Nicolas, Broutin, Luc, and Devroye, Eds., vol. AQ. Montreal, Canada: DMTCS, Jun. 2012, pp. 243–264. [Online]. Available: https://hal.inria.fr/hal-00666399
[26] W. A. Lorenz and Y. Ponty, "Non-redundant random generation algorithms for weighted context-free grammars," *Theoretical Computer Science*, vol. 502, pp. 177–194, 2013.
[27] R. Nussinov and A. B. Jacobson, "Fast algorithm for predicting the secondary structure of single-stranded RNA." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 77, pp. 6309–6313, Nov. 1980.
[28] D. H. Turner and D. H. Mathews, "NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure." *Nucleic acids research*, vol. 38, pp. D280–D282, Jan. 2010.
[29] M. Waterman, "Secondary structure of single-stranded nucleic acids," *Advances in Mathematics: Supplementary Studies*, vol. 1, pp. 167–212, 1978.
[30] M. Zuker and D. Sankoff, "RNA secondary structures and their prediction," *Bulletin of mathematical biology*, vol. 46, no. 4, pp. 591–621, 1984.
[31] Y. Ponty, "Efficient sampling of RNA secondary structures from the Boltzmann ensemble of low-energy: The boustrophedon method," *Journal of Mathematical Biology*, vol. 56, no. 1-2, pp. 107–127, 2008. [Online]. Available: https://hal.inria.fr/inria-00548863
[32] S. Will, K. Reiche, I. Hofacker, P. Stadler, and R. Backofen, "Inferring noncoding RNA families and classes by means of genome-scale structure-based clustering." *PLoS Comput. Biol.*, vol. 3, p. e65, Apr. 2007.
[33] J. N. Zadeh, B. R. Wolfe, and N. A. Pierce, "Nucleic acid sequence design via efficient ensemble defect optimization," *Journal of Computational Chemistry*, vol. 32, no. 3, pp. 439–52, 2011.
[34] J. Michalik, "Non-redundant sampling in RNA bioinformatics," Ph.D. dissertation, Université Paris-Saclay, Apr. 2019.
[35] S. Janssen and R. Giegerich, "The RNA shapes studio." *Bioinformatics (Oxford, England)*, vol. 31, pp. 423–425, Feb. 2015.
[36] ——, "Faster computation of exact RNA shape probabilities." *Bioinformatics (Oxford, England)*, vol. 26, pp. 632–639, Mar. 2010.
[37] M. Vingron and P. Argos, "Determination of reliable regions in protein sequence alignments." *Protein engineering*, vol. 3, pp. 565–569, Jul. 1990.
[38] E. Jacox, C. Chauve, G. J. Szöllsi, Y. Ponty, and C. Scornavacca, "ecceTERA: comprehensive gene tree-species tree reconciliation using parsimony." *Bioinformatics (Oxford, England)*, vol. 32, pp. 2056–2058, Jul. 2016.
[39] C. Chauve, J. Courtiel, and Y. Ponty, "Counting, generating, analyzing and sampling tree alignments," *International Journal of Foundations of Computer Science*, vol. 29, no. 05, pp. 741–767, 2018.

Christelle Rovetta has been a postdoc at LRI (Univ. Paris-Saclay) and at LIX (Ecole Polytechnique). She received her Ph.D. at Ecole Normale Supérieure de Paris, France in June 2017, following two Master of Science degrees in Applied Mathematics and Computer Science. Her main research interests are Simulation and Algorithms for Markov Chains analysis, RNA secondary structure, and machine learning.

Juraj Michalik Juraj Michalik is a bioinformatician, who defended a PhD in computer science from Ecole Polytechnique, France, following initial engineering studies at Institut National des Sciences Appliquées in Lyon, France. He currently holds a postdoc position at Institute of Molecular Biology, Czech Republic, where he studies the dependency of T-cell receptor sequences, their type and pathogen affinity.

Ronny Lorenz Ronny Lorenz is a bioinformatician who received his PhD in molecular biology in 2014. He currently holds a University Assistant position at the Department of Theoretical Chemistry, and is working on various aspects of RNA secondary structure prediction algorithms. Since 2010, he is the leading developer of the ViennaRNA Package. His main research interests are algorithmic aspects of RNA secondary structure prediction and the development of novel methods in RNA folding kinetics prediction.

Andrea Tanzer Andrea Tanzer is an Elise Richter fellow at the Medical University of Vienna at the Center for Anatomy and Cell Biology. She holds an MSc in Biology/Genetics from the University of Vienna and a PhD in Computer Science/Bioinformatics from the University of Leipzig. She was a member of ENCODE pilot and phase2, and Austrian PI of the RNALands consortium. Her research focuses on RNA bioinformatics and genomics, including big data analysis in transcriptomics, RNA folding kinetics, ncRNA detection/annotation, RNA structure prediction in vertebrates, plants and viruses, RNA G-quadruplex analysis and epitranscriptomics.

Yann Ponty Yann Ponty is a senior research scientist since 2021 at the French center for scientific research (CNRS), based at Ecole Polytechnique, where he currently leads the AMIBio research group in computational biology. He received his PhD and Habilitation in Computer Science from Université Paris-Saclay, has held postdoctoral positions at Boston College (USA) and Sorbonne University (France), and has been a visiting scientist at the Simon Fraser University (Canada). His main research interests include Discrete Mathematics and Algorithms applied to Bioinformatics, where he has contributed more than 80 manuscripts in journals and international conferences.