FPT Learning of Sparse, Robust and Interpretable Generative Models of RNA Evolution

Samuel Gardelle
LIX, CNRS, École polytechnique, Institut Polytechnique de Paris, Palaiseau, France

Laurent Bulteau
LIGM, CNRS, Université Gustave Eiffel, Marne-la-vallée, France

Yann Ponty
LIX, CNRS, École polytechnique, Institut Polytechnique de Paris, Palaiseau, France

Abstract

RNA structure modeling greatly benefits from the availability of structural homologs, associated with the presence of coevolving positions in multiple alignments. Direct Coupling Analysis (DCA) is a statistical framework for inferring significant covariations as Potts models, in a way that corrects for the transitive nature of mutual information. Various instances of DCA have been proposed over time with demonstrated ability to infer molecular contacts, yet were shown to be associated with inference algorithms that are invariably data hungry, prone to overfitting, and hindered by numerical instability. Recently, edge-activated DCA (eaDCA) has emerged as an alternative which iteratively infers couplings in a greedy manner and explicitly targets sparsity.

In this work, we revisit the inference of eaDCA models in a rigorous algorithmic setting. We circumvent the #P-hardness of computing the most promising addition/update of coupling and provide exact fixed-parameter tractable algorithms for the treewidth parameter of the coupling-induced graph. We empirically show that eaDCA models are typically associated with moderate treewidth values, and validate the practical feasibility of the method by producing, in a matter of minutes, the models associated with 41 RFAM families associated with structured non-coding families. Our results reveal good recovery rates for couplings associated with conserved base pairs from the family consensus, and enable a more systematic and robust assessment of the potential of eaDCA.

2012 ACM Subject Classification Applied computing → Molecular evolution; Theory of computation → Parameterized complexity and exact algorithms; Applied computing → Bioinformatics

Keywords and phrases RNA, Structure prediction, Evolution, DCA, Tree decomposition

Digital Object Identifier 10.4230/LIPIcs.WABI.2026.11

Supplementary Material *Software*: `https://codeberg.org/samuel-gardelle/wabi26-eaDCAstar` [6], archived at `swh:1:dir:e53ad07a87e78c5d6e48b33986c10a1c937ed706`

Funding *Samuel Gardelle*: École Normale Supérieure de Lyon

Acknowledgements The authors wish to thank Martin Weigt, Francesco Calvanese, Giovanni Peinetti, Andrea Pagnani, Ivo Hofacker and Sebstian Will for helpful discussions and advice.

1 Introduction

Context. *RiboNucleic Acids (RNAs)* are ubiquitous biomolecules that participate as first-class citizens in virtually every biological mechanism. They act both as medium to store genetic information, participate as enzymes in a variety of cellular functions (replication, regulation, sensing of metabolites...), and are widely-believed to represent a likely substrate for the origin of life [8]. Despite the prevalence of RNA transcripts in the Human genome [21], and the wide availability of RNA data produced by sequencing experiments (RNAseq) [12], our understanding of these molecules still remains partial. A workable insight into the inner workings of natural *RNA families* is essential to understand their biology, but also to inform a *rational design* of *synthetic RNAs*. The use of *therapeutic RNAs* in bio-medicine

is increasingly popular, and notoriously includes small antisense RNAs [15] and mRNA vaccines [18]. This motivates the development of design methodologies capturing statistical biases, induced by an *evolutionary pressure* to preserve RNA *function*.

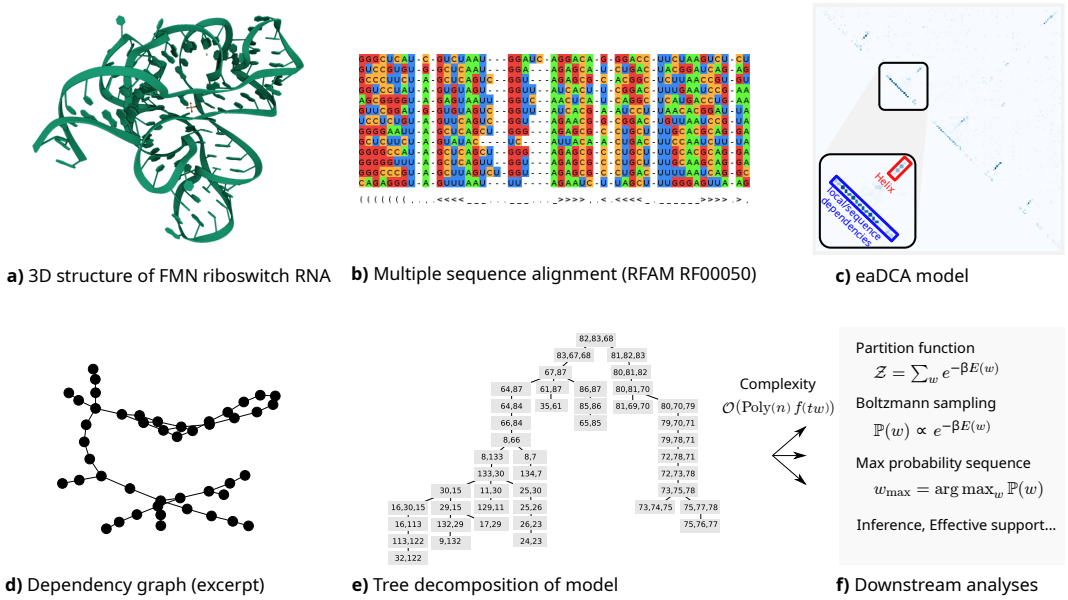
Although RNA is presented as a sequence of *nucleotides*, chemical units abstracted as A, C, G or U, the *structure* adopted by RNA is often a major determinant of its function, e.g. enabling its recognition by molecular partners. Functionally-similar RNAs, also called *homologs* [5], are often found within evolutionary-distant organisms, indicating their widespread relevance. This induces an evolutionary pressure with statistical consequences at the *nucleotide* sequence level. Notably, *base pairs*, mediated by hydrogen bonds, involving two sites will restrict their content to pairs of *complementary* nucleotides. Such a *co-evolution* can be revealed by the analysis of the mutual-information [4], informed by adequately-constructed *Multiple Sequence Alignments (MSAs)* [10].

Direct Coupling Analysis (DCA) [19] is a popular paradigm for an – information based – analysis of biomolecules and, more specifically, RNA families [4]. It posits that the evolutionary pressure on an RNA family can be modeled as a *Potts model*, *i.e.* a set of functions associating a weight, akin to an energy, to the content of a pair of sites (*coupling*). Such a model then induces a *Boltzmann-Gibbs distribution* on the entire space of RNA sequences (of a given length), leading to *generative models* that have been used for *molecular modeling and design* [3]. DCA models are typically inferred under the dual objective of inducing single and dual site statistics that best approximate the ones observed in MSAs, while maximizing the joint *emission probability* for RNA sequences in the family (*Maximum Likelihood Estimation; MLE*). DCA is useful to analyze co-evolution because couplings directly model co-evolving positions, e.g. strong couplings that support compatibility can typically be interpreted as evidence of proximity/connectedness on a structural level. Moreover, it has been shown experimentally [14] that it can be useful for the design of new functional RNAs.

The *interpretability* of DCA is further boosted by the typical *parsimony* of produced models, which results from DCA explicitly avoiding transitive information, and is guaranteed by design in the context of *edge-activated DCA (eaDCA)* [3]. This recently proposed version of DCA has been shown to achieve similar performances as earlier propositions, while only allowing a subset of iteratively-added edges (some possibly removed along the way) to contribute, until a predefined fraction of the single and pair statistics is induced by the model. Edges that are initially activated by eaDCA are typically part of the RNA structure, mainly located in the secondary structure, but also reveal more complex dependencies such as negative constraints preventing the adoption of alternative stable structures. However, the computational methodologies used to infer eaDCA models, sample sequences and estimate additional relevant features (e.g. *partition function*, effective support size) heavily rely on heuristics (stochastic optimization procedures; *MCMC*). Indeed, developed in the context of statistical physics/biophysics, such approaches generally do not offer guarantees of convergence, which has been identified by authors as the root cause for a loss of scalability, precision, *stability* and/or reproducibility [3].

Contributions. In this work, we exploit the *sparseness* of eaDCA models to design exact algorithmic methods, through the lens of *parameterized complexity*, to perform *interpretable* Machine Learning and, ultimately, decipher the key determinants of RNA molecular evolution. Our contributions include:

- Designing exact dynamic programming algorithms, having complexity parameterized by the *treewidth*, to compute the partition function and perform *statistical sampling* (*aka.* weighted random generation) in the Boltzmann-Gibbs distribution induced by a given eaDCA model;



a) 3D structure of FMN riboswitch RNA **b)** Multiple sequence alignment (RFAM RF00050) **c)** eaDCA model

d) Dependency graph (excerpt) **e)** Tree decomposition of model **f)** Downstream analyses

Figure 1 Molecular modeling with eaDCA and tree decomposition-based algorithms. To determine the (3D) conformation of RNA (a), co-evolving pairs of positions are extracted from a multiple sequence alignment of homologous RNA sequences (b), leading to an evolutionary coupling whose content reveals various structural features (c). Couplings can be seen as an undirected graph (d), whose tree decomposition, typically in combination with *dynamic programming*, allows to compute relevant metrics, and execute key (*NP-hard*) analyses, in time only exponential in the treewidth (f).

- Revisiting the *inference* of eaDCA models in an attempt to improve their efficiency, stability and representativity of specific RNA families;
- Assessing, in a more systematic manner, the capacity of eaDCA to predict key structural features at both the secondary and tertiary structure level, but also identify negative evolutionary pressure and, possibly, alternative structures.

2 Context and problems statement

Direct Coupling Analysis (DCA) is a statistical modeling paradigm for molecular families. A DCA *model* (also known as Potts models, or binary Markov networks) is based on a multiple sequence alignment of the family it attempts to model. It infers an energy function (ie. a weighting) for the space of aligned sequences $\{A, C, G, U, -\}^N$ where $\Sigma = \{A, C, G, U, -\}$ is the alphabet, $-$ denotes insertions/deletions and $N \in \mathbb{N}$ is the length of the alignment. Ideally, a *good* DCA model assigns low energies to sequences that share a common function/structure as the family, and high energies to non-functional sequences.

The energy function $E(w)$ is required to have the following form

$$-E(w \in \Sigma^N) = \sum_{1 \le i \le N} h_i(w_i) + \sum_{1 \le i < j \le N} J_{i,j}(w_i, w_j).$$

where w_k denotes the k-th letter of the sequence w. The functions $(h_i)_{1 \le i \le N}$ are known as *local fields*, and $(J_{i,j})_{1 \le i < j \le N}$ as *couplings*. In the worst case scenario, there could be as much as $\Theta(N^2)$ couplings yet, in practice, few of them are non-zero. Non-zero couplings are known as *activated* couplings, and we refer to models with few activated couplings as *sparse*.

Two fundamental, mutually dependent, tasks in DCA modeling are:

- *Learning* a model from an alignment, by choosing the local fields h_i and the couplings $J_{i,j}$ so as to: i) reproduce nucleotide distributions for individual sites and pairs thereof; and ii) maximize the joint emission of sequences in the training set;
- *Sampling* from the *Boltzmann* distribution induced by the inferred energy model:

$$P(w) = \frac{e^{-E(w)}}{Z(w)}$$

with

$$Z = \sum_{w \in \Sigma^N} e^{-E(w)}$$

a normalizing constant known as the *partition function*.

This distribution assigns high (resp. low) probability to low-energy (resp. high-energy) sequences, and can be used to design RNAs with similar function as those of the model family [4, 3]. Unfortunately, computing the partition function under general couplings is #P-hard, as DCA couplings can be set to emulate Ising models, for which computing the partition function is #P-Complete [9].

The edge-activated DCA (eaDCA) framework

Edge-activated Direct Coupling Analysis (eaDCA), introduced by Calvanese *et al.* [3], is a recent installment of DCA, which focuses on learning sparse models in a greedy manner. It adheres to the well-established framework of *maximum likelihood learning* used in statistical learning. In this framework, the likelihood function is

$$L(M; \text{dataset}) = \prod_{x \in \text{dataset}} P_M(x) = \prod_{x \in \text{dataset}} \frac{e^{-E(x)}}{Z}$$

where $P_M(x)$ is the probability of x in the model $M = (J, h)$. It represents a measure of how well the model fits the data.

Within the *eaDCA* framework, the inference starts with an empty model, with no activated coupling, and using local fields $\{h_i\}_{1 \leq i \leq N}$ chosen to mimic site statistics. It then iteratively activates new couplings in a greedy manner, choosing at each step the coupling (and associated functions values) which maximizes the likelihood of the model. Fig. 2 details the criteria used to select new couplings along with their values, and when to stop the iteration.

Using this approach, its authors were able to produce sparse, yet accurate, models on a selection of RNA families [20]. Moreover, those results were obtained while activating a limited proportion of couplings, typically $\Theta(N)$ out of $\Theta(N^2)$, which turned out to substantially overlap with base pairs occurring in the structural consensus of the family.

The sparsity shown by eaDCA models is desirable in that it enables *interpretability* and *knowledge discovery*. A simple example of interpretable parameters is that of couplings $J_{i,j}(x, y)$ assigning large negative values when x and y are complementary nucleotides, and big positive values otherwise. Such couplings promote the pairing of columns i and j, and suggest the existence of a corresponding evolutionary pressure towards maintaining a base pair in the column. Such couplings frequently involve pairs of columns that are backbone-adjacent, part of the consensus structure or, more generally, proximate in 3D models [3].

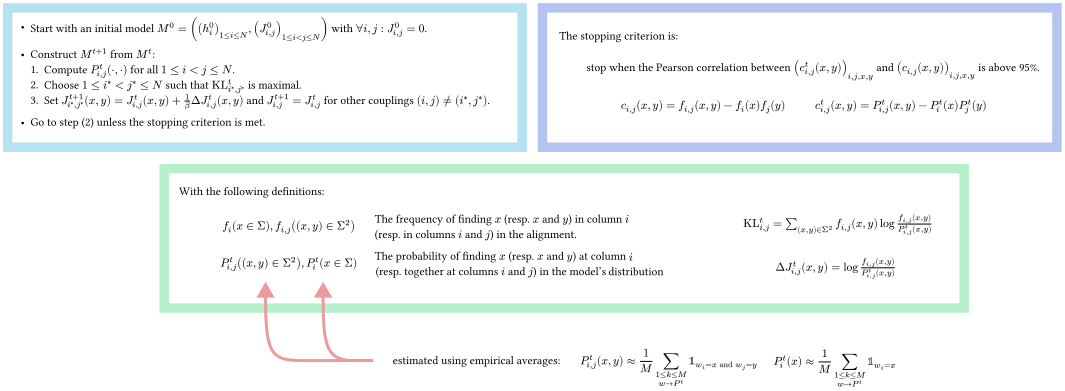


Figure 2 Greedy maximum-likelihood learning in the style of eaDCA [3].

Back to learning, the original implementation of eaDCA heavily relies on *Markov Chain Monte Carlo (MCMC)* to approximately sample the distributions of intermediate models. These samples are used to compute estimates of marginals via empirical averages, a key step in the main loop of the inference algorithm (Figure 2). Such an approach represents an important shortcoming of the method: since the implementation of MCMC as it is used does not provide convergence bounds, approximate sampling is done with an unknown approximation error, and it follows that the precision of frequency estimates cannot be guaranteed. Moreover, the mixing time of the underlying Markov Chain, exponential in a worst-case analysis, is practically too long so the original designer of eaDCA ended up intentionally rely on out-of-equilibrium statistics whenever consequential couplings are added. Those limitations impede stability, and possibly reproducibility since the method is inherently random, the returned model structure depends on the seed provided to the random number generator.

3 Treewidth as a stronger and tractable notion of sparsity

The sparsity of eaDCA models suggests the use of exact algorithmic approaches to improve the computation. Namely, we now set out to show that the greedy approach can be implemented *exactly* with reasonable time complexity under sparsity assumption. By *exact*, we mean that the quantities of Figure 2 are computed with no other error except from those arising from floating-point operations, we don't mean that the learning procedure finds the maximum-likelihood model, indeed the greedy nature of the algorithm doesn't guarantee convergence to the maximum.

Towards that goal, we set out to design parameterized algorithms, based on the notion of *treewidth* of a graph, to solve the problem. The treewidth is a structural graph parameter which intuitively measures how *tree-like* a graph is, achieving low values for graph reduced to paths and trees, and larger $\Theta(n)$ values for graphs that are (near) complete. While formally divergent from the notion of sparsity, the treewidth is increasing upon adding edges, and typically correlates with the notion of sparsity. Graphs with small treewidth have a structure that can be exploited algorithmically, enabling an exact resolution of many NP-hard problems in time that grow polynomially on the size N of the instance, and only super-polynomially (*e.g.* exponentially) on the treewidth tw of the input graph. Such *Fixed-Parameter Tractable (FPT)* problems include Independent-Set and 3-SAT, both of which can be solved in time $O(\alpha^{tw} \cdot P(N))$ with α a constant and P a polynomial.

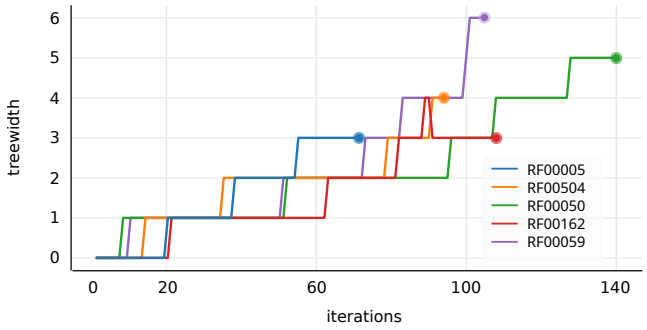


Figure 3 Empirical treewidths of the eaDCA models produced by Calvanese *et al.* [3]. The learning procedure iterates a greedy strategy which stops when 95% of the single site and pair statistics are explained by the model. For all analyzed multiple sequence alignment, each associated with an RFAM families, the treewidth tw of the final model remained modest (3 to 6), making $\mathcal{O}(n.|\Sigma|^{tw})$ DP-based FPT algorithms an attractive alternative to the unstable numerical procedures (MCMC) initially proposed by its authors.

As shown in Fig. 3, we observe that the graph of activated couplings, induced by eaDCA models learned by greedy maximum likelihood, have systematically low treewidth. This motivates the design of tractable and exact algorithms, as follows:

▶ **Theorem 1** (Parameterized greedy learning). *Given an alignment of n sequences of length N, the maximum-likelihood greedy DCA model M^T after T steps can be obtained in time $O(nN^2 + tw^2 5^{tw} N^2 T + TC(tw; N))$ where:*

- *tw is the treewidth of the graph of activated couplings of the model M^T;*
- *$C(tw; N)$ is the complexity of computing tree decompositions for graphs of treewidth tw.*

▶ **Theorem 2** (Parameterized sampling). *Given a DCA model M of length N along with a tree decomposition of the graph of its activated coulplings of width tw, we can draw a sample from its Boltzmann distribution in time $O(tw^2 5^{tw} N)$.*

It follows from the celebrated results of Bodlaender [2] that $C(tw; N) \in \mathcal{O}(N.2^{\mathcal{O}(tw^3)})$, therefore eaDCA learning is FPT for the treewidth. In practice, the cost C of computing tree decompositions is negligible compared to the other time complexities.

▶ **Corollary 3.** *Greedy maximum-likelihood learning and sampling of DCA models are Fixed-Parameter Tractable for the treewidth of those models.*

Those algorithms are all based on the use of tree decompositions, that we now define:

▶ **Definition 4** (Tree decomposition). *For an undirected graph $G = (V, E)$, a tree decomposition of G is a tuple $\mathcal{T} = (T, (X_v)_{v \in V(T)})$ where T is a directed tree and $(X_v)_{v \in V(T)}$ is a family of subsets of $V(G)$ indexed by nodes of $V(T)$. The elements of $(X_v)_{v \in V(T)}$ are called the bags of the decomposition. Moreover, a decomposition must satisfy the following axioms:*

1. $\cup_{v \in V(T)} X_v = V(G)$ *(bags cover all the graph's vertices)*
2. $\forall (i, j) \in E(G), \exists v \in V(T) : i \in X_v$ *and* $j \in X_v$ *(every edge of the graph is covered by a bag)*
3. *if $v, w \in V(T)$ are two nodes of the decomposition and $x \in V(G)$ is a node of the graph such that $x \in X_v, x \in X_w$ then any node $u \in V(T)$ on the path between v and w must be such that $x \in X_u$. (running intersection property)*

Finally, the width *of the decomposition $\mathcal{T}$ is defined as $max_{v \in V(T)} |X_v| - 1$.*

The treewidth of a graph is the width of its decomposition of smallest width, thus a tree decomposition of width tw certifies that its associated graph has treewidth at most tw. It gives an upper bound on how fast dynamic programming algorithms using this decomposition will run.

Although it is NP-Hard to compute the treewidth, or a tree decomposition of smallest width for a general graph [1], we can assume, as argued before, that we can easily obtain decompositions of small width for the graphs we are interested in. It is indeed what we observe in practice.

Moreover, we will also assume that those decompositions have size $O(N)$. This fact is guaranteed by our choice of heuristic used to compute them, but in general we can reduce the size of any bigger decomposition to at most $3N = O(N)$ vertices for T [13].

3.1 Computing the partition function of DCA models

Now that we have presented the framework of tree decompositions, we show how it can be used to compute the *partition function* of DCA models. The partition function is not directly useful for our purpose, but is an important quantity in statistical physics, and once we have access to it, we can perform the useful tasks of learning or sampling of the models.

For a DCA model $M = ((h_i)_{1\leq i\leq N}, (J_{i,j})_{1\leq i<j\leq N})$, the graph induced by its activated couplings is $G_M = (\{1,..,N\}, E(G_M))$ where $\{i,j\} \in E(G_M) \iff J_{i,j}(\cdot,\cdot) \neq 0$. In the rest of the paper, we will assume that we have access to $\mathcal{T} = (T, (X_v)_{v\in V(T)})$ a tree decomposition of G_M of low treewidth, along with a partition $f : V(T) \to \{J_{i,j}\}$ of the activated couplings of M:

$$\{J_{i,j} \neq 0\} = \bigsqcup_{v\in V(T)} f(v)$$

Such a partition is guaranteed to exist by the second axiom of tree decompositions (see definition 4), and can be found efficiently in time $O(tw^2 N)$. Also for the sake of concision, we will ignore the local fields $(h_i)_{1\leq i\leq N}$ in our algorithmic treatment. Those can be handled by extending the theory with loops $J_{i,i} = h_i$.

Moreover, we will use the following notations:

- $D_v \subset V(T)$ are the descendants of node v (included), in T.
- $Y_v = \cup_{u\in D_v} X_u$
- $X_{pa(v)} \subset \{1,..,N\}$ is the bag X_w for w the parent of v if it exists or $\emptyset$ if v has no parent.
- For $s \in \Sigma^A, s' \in \Sigma^B$ with $A \subset B$, $s \subset s'$ means that s' agrees with s on its domain.

We can define the partition function:

▶ **Definition 5** ((generalized) partition function).
Let $v \in V(T)$ be a node of $\mathcal{T}$, the partition function Z_v is the function:

$$Z_v(s \in \Sigma^{X_v \cap X_{pa(v)}}) = \sum_{\substack{s'\in\Sigma^{Y_v} \\ s\subset s'}} e^{-\sum_{u\in D_v}\sum_{J_{i,j}\in f(u)} J_{i,j}(s(i),s(j)))} = \sum_{\substack{s'\in\Sigma^{Y_v} \\ s\subset s'}} e^{-E_v(s)}$$

with $E_v(s) = \sum_{u\in D_v}\sum_{J_{i,j}\in f(u)} J_{i,j}(s(i), s(j))$.

Note that when v is the root we recover $Z_v(\emptyset) = Z = \sum_{s\in\Sigma^N} e^{-\sum_{J_{i,j}} J_{i,j}(s_i,s_j)}$ because $D_v = V(T)$ and f is a partition of M's couplings. For a general $v \in V(T)$, the quantity $Z_v(s)$ can be interpreted as the weight of the assignments s' that extend s in the Boltzmann distribution, when considering the submodel composed of columns/couplings below v in $\mathcal{T}$.

The key lemma that justifies the use of tree decompositions is the following. It provides a recursive definition of partition functions that can be used as a dynamic programming equation:

▶ **Lemma 6** (DP equation for the partition function). *Let $v \in V(T)$ be a vertex of the decomposition such that: $u_1, .., u_k$ are the direct children of v in T. Then the following holds:*

$$Z_v(s \in \Sigma^{X_v \cap X_{pa(v)}}) = \sum_{\substack{s' \in \Sigma^{X_v} \\ s \subset s'}} e^{-\sum_{J_{i,j} \in f^{-1}(v)} J_{i,j}(s'(i), s'(j))} \times \prod_{1 \leq i \leq k} Z_{u_i}(s'|_{X_{u_i} \cap X_v})$$

where $s|_S$ is the restriction of s onto the columns of S.

Proof. (available in the appendix) ◀

A dynamic programming algorithm implementing this formula has time complexity:

$$O\left(\sum_{v \in V(T)} |X_v \cap X_{pa(v)}| \times |X_v - X_{pa(v)}| \times |f^{-1}(v)| \times deg(v)\right) = O(5^{tw+1} tw^2 N)$$

3.2 Sampling from the Boltzmann distribution of a DCA model

Access to the partition functions enables us to sample the model directly through the following algorithm:

▶ **Lemma 7.** *Let $v_1, .., v_n$ be a reversed topological ordering of T starting from the root. The following algorithm samples from the Boltzmann distribution of M:*

Let s be the unique assignment of $\Sigma^{\emptyset}$.
for *$1 \leq i \leq n$* ***do***
  Let $B = X_{v_i} - dom(s)$.
  Sample $s' \in \Sigma^B$ with probability proportional to $Z_{v_i}((s + s')_{|X_{v_i} \cap X_{pa(v_i)}})$.
  Extend s with the values of s'.
end for

Proof. (available in the appendix) ◀

The sample s is built from a partial assignment of $\Sigma^{\{1,..,N\}}$ whose domain is gradually populated with the variables present in the sets X_v of the nodes $v \in V(T)$ that are visited. The intermediate samples are drawn from a distribution whose values depend entirely on Z_v and s, showcasing that the hard part of this sampling algorithm is the computation of partition functions.

3.3 Computing pairwise marginals of the Boltzmann distribution

The marginals $P_{i,j}(x, y)$ for $1 \leq i < j \leq N$ and $x, y \in \Sigma$ are central to the greedy learning algorithm. In a similar way to the original eaDCA paper [3], we may estimate those marginals by using samples from the Boltzmann distribution (Lemma 7), this time with the guarantee that the sampling is not approximate. But it is in fact possible to remove the randomness entirely, using the technique that we now describe.

To compute the marginals, we need to have access to an additional type of partition functions known as an outside partition function:

▶ **Definition 8** (outside partition function). *Let $v \in V(T)$ be a node of $\mathcal{T}$, the outside partition function Z_v^o is the function:*

$$Z_v^o(s \in \Sigma^{X_v \cap X_{pa(v)}}) = \sum_{\substack{s' \in \Sigma^{Y_v} \\ s \subset s'}} e^{-\sum_{u \in V(T)-D_v} \sum_{J_{i,j} \in f(u)} J_{i,j}(s'(i),s'(j)))} = \sum_{\substack{s' \in \Sigma^{Y_v} \\ s \subset s'}} e^{-E_v^o('s)}$$

with $E_v^o('s) = \sum_{u \in V(T)-D_v} \sum_{J_{i,j} \in f(u)} J_{i,j}(s'(i), s'(j))$. The only change here is that the sum is done over $u \in V(T) - D_v$.

Similarly, outside partition functions can be computed in the same time complexity using the following dynamic programming equation:

▶ **Lemma 9.** *Let $v \in V(T)$ be a node of the decomposition such that $u_1, .., u_k$ are the direct children of v in T. Then the following holds $\forall 1 \leq l \leq k$:*

$$Z_{u_l}^o(s \in \Sigma^{X_{u_l} \cap X_v}) = \sum_{\substack{s' \in \Sigma^{X_v} \\ s \subset s'}} e^{-\sum_{J_{i,j} \in f^{-1}(v)} J_{i,j}(s'(i),s'(j))} \times Z_v^o(s'_{|X_v \cap X_{pa(v)}}) \times \left(\prod_{\substack{1 \leq i \leq k \\ i \neq l}} Z_{u_i}(s'|_{X_{u_i} \cap X_v}) \right)$$

Proof. (omitted, similar proof to Lemma 6: decompose the non-descendants of u_l as its parent v, its siblings' descendants, and its parent's non-descendants.) ◀

The main differences here being that the computation is performed starting from the top of the tree T towards the leaves, and that both types of partition functions are used.

It can be observed that $E_v(s) + E_v^o(s) = E(s)$ because $V(T) = D_v \sqcup (V(T) - D_v)$ suggesting that the two types of partition functions decompose the model into two parts. Due to the separation properties of tree decompositions (see Lemma 12 in the appendix) this intuition extends further with the following lemma:

▶ **Lemma 10.** *Let $v \in V(T)$ and P_v be the marginal of the Boltzmann distribution restricted to the columns of $X_v \cap X_{pa(v)}$. Then for $s \in \Sigma^{X_v \cap X_{pa(v)}}$ we have that:*

$$Z_v^o(s) \times Z_v(s) = \sum_{\substack{s' \in \Sigma^{\{1,..,N\}} \\ s \subset s'}} e^{-\beta E(s')} \propto P_v(s)$$

Proof. (available in the appendix) ◀

This provides us with a way to compute some of the marginals of the Boltzmann distribution. Namely, those restricted to $X_v \cap X_{pa(v)}$ for some $v \in V(T)$. If a pair of columns $\{i, j\}$ is included in one of them then we can compute $P_{i,j}$ by summing out the other variables. Indeed, letting $A \subset B \subset \{1, .., N\}$ we have from probability theory that for any distribution Q: $Q(s \in \Sigma^A) = \sum_{\substack{s' \in \Sigma^B \\ s \subset s}} Q(s')$. However, for a fixed tree decomposition there can only be at most $O(tw^2 N)$ such pairs $\{i, j\}$ while we need to compute $\Omega(N^2)$ of them. To circumvent this problem, we can construct decompositions $\mathcal{T}^{i^*}$ for $1 \leq i^* \leq N$ from $\mathcal{T}$ such that all the edges in $(\{i^*, j\})_{1 \leq j \leq N}$ are such that there exists $v \in V(T^{i^*})$ with $\{i^*, j\} \subset X_v \cap X_{pa(v)}$.

Algorithm 1 Compute pairwise marginals of a DCA model given a tree decomposition $\mathcal{T}$.

for $1 \leq i \leq N$ **do**
  Compute $\mathcal{T}^i$ from $\mathcal{T}$ in time $O(N(tw+1))$.
  Compute Z_v for every $v \in V(T^i)$ in time $O(5^{tw+1}tw^2N)$.
  Compute Z_v^o for every $v \in V(T^i)$ in time $O(5^{tw+1}tw^2N)$.
  for $v \in V(T^i)$ **do**
    Let $P_v(s) = Z_v(s) \times Z_v^o(s)$.
    for $\{i,j\} \subset X_v$ with $i < j$ **do**
      if $P_{i,j}$ has not been computed yet **then**
        Compute $P_{i,j}$ from P_v by summing out $X_v - \{i,j\}$.
      end if
    end for
  end for
end for

▶ **Lemma 11** (Augmented tree decompositions). *Let $\mathcal{T}$ be a tree decomposition of G of size $s \in \mathbb{N}$ and $x \in V(G)$. We can build a decomposition $\mathcal{T}^x$ of size $2s$ and treewidth at most $tw(\mathcal{T})+1$ such that for every $y \in V(G)$ there exists $v_y \in V(T^x)$ such that:*

$$\{x,y\} \subset X_{v_y} \cap X_{pa(v_y)}$$

Proof. (available in the appendix) ◀

Using this lemma and the rest of the techniques that we have introduced previously, we can compute the marginals $P_{i,j}$ for all $1 \leq i < j \leq N$ as shown if Algorithm 1 in time $O\left(5^{tw+2}N^2tw^2\right)$.

This completes the proof of Theorem 1 if we use the marginals $P_{i,j}$ to implement T steps of eaDCA as in Figure 2, and we take into account the pre-computation of the statistics $f_{i,j}, f_i$ in time $O(nN^2)$ as well as the cost of finding tree decompositions.

4 Empirical analyses

The ability of eaDCA to effectively model RNA families was already investigated in the original paper [3] on a limited selection of families associated with large alignments. However, our FPT alternative enables the assessment of the method at a larger scale, and test various hypotheses on the behavior of the framework. To that purpose, we implemented our exact treewidth-based revisiting of eaDCA as a `Rust` software named `eaDCA`*. The code, data and detailed per-family results associated with these experiments are available at `https://codeberg.org/samuel-gardelle/wabi26-eaDCAstar`.

Datasets. We consider family-level multiple sequence alignment, downloaded from the 15.1 version of RFAM [20]. We used the *seed alignments* of families, usually established through an expert-driven combination of computational methods and manual curation.

Out of the 4 227 RFAM families currently in RFAM, 41 were selected (see full list in Appendix Table A.1) to match the following criteria:

1. The (horizontal) alignment length must be between 100 and 400 nts;
2. The alignment should include at least 100 sequences;
3. The average pairwise Hamming distance should not exceed 80%;
4. The consensus structure of the family should be tagged as "Published" (*vs* "Predicted");
5. At least 50% of positions should be paired.

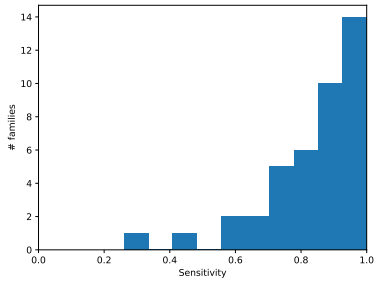


Figure 4 Distribution of sensitivity, *i.e.* proportion of base pairs from the RFAM consensus structure that are activated as couplings by `eaDCA`*.

The length requirement (Criterion 1) is mainly meant to: i) avoid smaller RNA families, such as the 1 500+ miRNA precursors associated with seed alignments of lower quality; as well as ii) larger alignments resulting from horizontal expansion (gaps), indicating lower alignment quality while inducing unreasonable computational demands. Criterion 2 and 3 represent minimal prerequisites for the alignment (and underlying RNA family) to showcase some level of sequence-level covariation. We require some level of expert scrutiny over the consensus structural model (Criterion 4). Finally, we forced the presence of a critical proportion of base pairs (Criterion 5), hopefully indicative of a family undergoing evolutionary pressure at a structural level.

Experiments. We conducted further computational experiments, by systematically learning models for the families of the RFAM database. We used the alignments during the learning phase to compute the frequencies $f_{i,j}(\cdot,\cdot)$ and $f_i(\cdot)$ using a pseudocount estimator ($\alpha = 1$). For each family, we ran `eaDCA`* to infer $T = 200$ couplings, prematurely stopping whenever the graph induced by the current couplings attains a treewidth exceeding 5. The evolution of the treewidth is depicted in Figure 9 in appendix.

Note that couplings are selected based on coevolutionary metrics alone and, in particular, do not necessarily reflect the capacity of two columns to form base pairs. In order to better distinguish couplings that are consistent with base pairing rules, we define the *Pairwise-Compatibility (PC)* score as

$$PC(i,j) = \frac{\sum_{\{x,y\}\in\{\{G,C\},\{U,A\},\{G,U\}\}} f_{i,j}(x,y)}{\sum_{x\neq- \text{ and } y\neq-} \sum_{x,y\in\Sigma} f_{i,j}(x,y)}.$$

For two unrelated, independent columns in an alignment, with no gaps the PC is expected to be $\frac{6}{16} = 0.375$.

The results, detailed henceforth, show that the method is applicable to a wider range of families. Indeed, we confirmed that `eaDCA`* is able to recover most basepairs present in the RFAM consensus structure. Moreover, the presence of couplings in the model correlate highly with small distances in PDB structures. Both of these points show the ability of DCA models to highlight positive evolutionary pressures, such that the presence of basepairs. We were able to go further, and show that some couplings suggest the presence of negative evolutionary pressures preventing e.g. stack extension, helix recombination, or the formation of additional structures in unpaired regions.

4.1 Testing the capacity of eaDCA* to recover RFAM consensus structures

We executed eaDCA* on the 41 RFAM families as described above on a domestic laptop, and obtained the couplings in less than a minute for each of the families. In every case, in the first 10 seconds we already obtained 100 couplings and a treewidth $tw \leq 5$. Details on the final number of couplings and the final treewidth can be found in the appendix.

Firstly, for a given RFAM family, we assess the capacity of eaDCA* to recover the set B of base pairs found in the consensus structure. Towards that goal, we compute the *sensitivity* of a model M, activating a list E_M of couplings, as $\text{Sens}(M) := \frac{|B \cap E_M|}{|B|}$.

As can be seen in Fig. 4, which shows the distribution of the sensitivity metrics, for 27 out of the 41 families, the models inferred by eaDCA* achieve sensitivities of 80% or more, and only 2 models feature less than 50% of base pairs in the RFAM consensus. This is reassuring, yet not entirely surprising since consensus structures are usually built from a specific type of co-evolutionary signal (covariance models [5]). Still, such performances are remarkable as, for many families, the number of base pairs in the consensus is close to the number of activated couplings. Moreover, consensus structures are usually drafted from a combination of evolutionary and physics-based evidence [7], enabling a prediction of base pairs that do not exhibit any meaningful coevolution (*e.g.* over fully conserved columns). Overall, we observe that eaDCA is generally able to prioritize couplings associated with compensatory mutations.

We illustrate in Fig. 5 the behavior of eaDCA* on six hand-picked RNA families. A visual inspection suggests that eaDCA* is able to recover most of the consensus base pairs, usually very early across iterations. Notably, the method does not appear to be hindered by the presence of crossing interactions/pseudoknots (see RF02996). We observe negative couplings, activated but associated with incompatible pairs of nucleotides (low PC value/red), are often found at the ends of helices, seemingly preventing their extension (see Subsection 4.2). Negative coupling can also be found within single regions that are predicted to remain unpaired (see RF00020), or across unpaired regions, seemingly as the result of an evolutionary pressure against their interaction (see RF02944 and RF02968).

Note that typical structural analyses consider some measure of specificity, such as the PPV, in addition to sensitivity. However, such an assessment is delicate as detecting structurally-conserved base pairs is not the sole goal of eaDCA. Indeed, it is known that activated couplings also frequently include neighboring positions on the sequence and 3D levels, as well as evidences, discussed below, of a negative selective pressure, acting against structurally-deleterious mutations. Moreover, any reasonable assessment of specificity would require a well-calibrated stopping criterion, an aspect that leave to future work.

4.2 Revealing negative evolutionary pressure

We further analyzed couplings involving incompatible columns, in an attempt to reveal negative evolutionary signals in the couplings that were inferred in the previous experiment. Towards this goal, we refine the notion of *disruptive base pairs* introduced by Yao *et al.* [23], and classify the couplings of eaDCA* models depending on their position in the consensus structure.

We distinguish 3 cases that are believed to correspond to negative evolutionary constraints:

- *Stack extensions* are couplings whose positions are present directly above or below an helix of the consensus structure. We also transitively include in that definition couplings that are below/above other stack extensions. Depending on the compatibility of columns, measured by the PC metrics, we posit that such couplings reflect a selective pressure against the further extension of an helix;

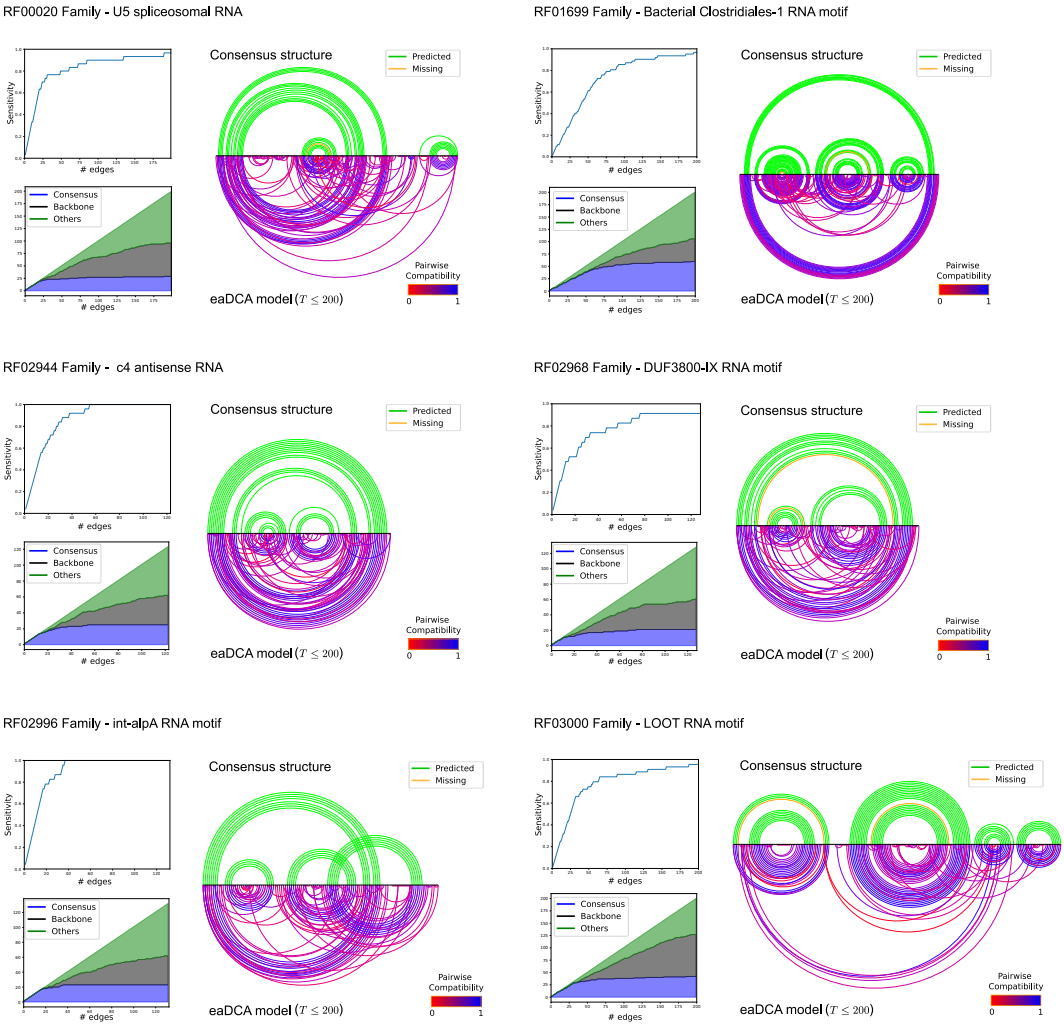


Figure 5 RFAM consensus structures and typical eaDCA models. For each RFAM family, we report the base pairs contained in the consensus structure (top-right) and those inferred by the couplings (bottom-right). Consensus base pairs are color-coded to illustrate their presence in (green) or absence from (orange) the consensus, and eaDCA couplings are drawn using a gradient from red to blue to indicate the proportion of compatible base pairs found in the alignment. The evolution, across iterations, of the sensitivity – the proportion of the consensus BPs found in the eaDCA model – is represented in the top left. Finally, we report (bottom-left), a breakdown of couplings into base pairs that are in the family consensus (blue), proximate in the backbone (black) or others (green).

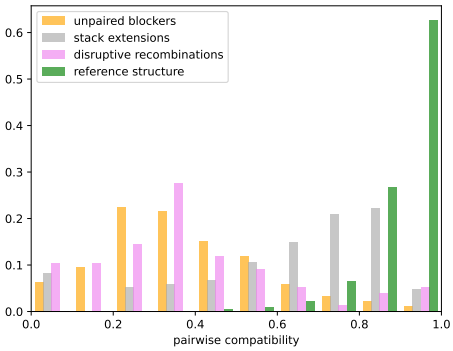


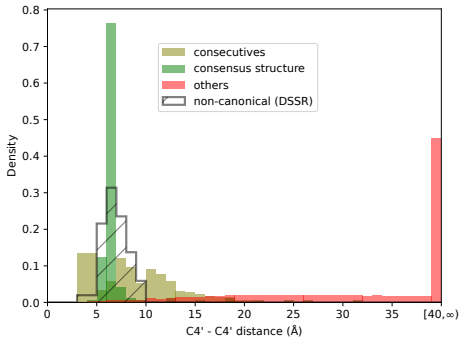
Figure 6 Distribution of pairwise compatibility (PC) among couplings, stratified based on their relationship with the consensus structure. Unpaired blocked couplings are contained within an unpaired region; stack extensions represent possible base pairs that would stack onto an existing helix; disruptive recombinations mediate (or forbid) interaction between two distinct helices.

- *Unpaired blockers* are couplings that involve two positions present in the same stretch of unpaired positions in the consensus structure. We posit that their presence reflects a loss of fitness incurred by the formation of additional helices, in a region whose accessibility is functionally important;
- *Disruptive recombinations* are couplings whose positions are present in the unpaired regions of two distinct hairpins. We posit that such couplings indicate an evolutionary pressure against the formation of a competing structure including the coupling.

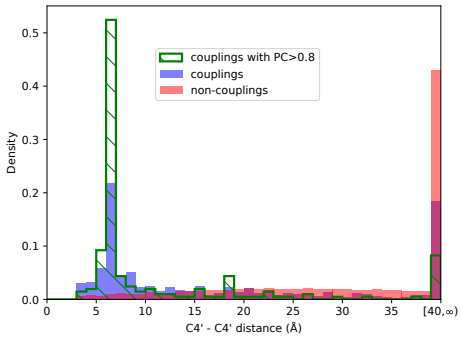
All of these mechanisms require that the associated couplings favor non-complementary base pairs to hinder the formation of base pairs. In other words, we expect the associated couplings to have low PC score. Figure 6 reports the distribution of PC values for the 3 categories of negative evolutionary couplings inferred by `eaDCA`*. For reference, we also report the PC distribution for couplings occurring in the consensus structure.

As expected, a clear divergence is observed between the consensus pairs and the negative couplings, the latter generally achieving much lower compatibility with respect to the PC metrics. An exception to that trend is observed for stack extensions. This is not overly surprising, since consensus structures within RFAM are predicted using diverse (at least partly manual) methodologies, some of which can be overly conservative especially within columns featuring gaps. We believe that such base pairs, having good mutual information, compatibility, and stacking onto an existing helix, could probably be safely added to the RFAM consensus structure.

Unpaired blockers and disruptive recombinations appear to induce PC distributions, with a low median value around 0.3. This moderate value is interesting, as it does not prevent the formation of base pairs in every sequence. It suggests that the repression of helices in unpaired regions may induce pervasive selection of limited intensity, inducing disjunctive incompatibility constraints, in a non-targeted manner and in combination with other local constraints. A notable difference can however be observed for extreme PC values (close to 1), where unpaired blockers are virtually absent, while 6% of disruptive recombinations achieve a PC close to 1. We hypothesize that this behavior may reflect the existence of alternative structures which would be important, and whose base pairs would therefore be positively selected by evolution.



(a) Distance distribution for pairs of columns in RFAM MSA: (Green) Base pairs found in the RFAM consensus structure; (Gold) Pairs (i, j) of quasi-consecutive positions ($j \leq i + 3$); (White/-Dashed) Pairs annotated as non-canonical base pairs in at least one 3D structure; (Red) Other columns pairs (background).



(b) Distance profile among `eaDCA`*-inferred couplings (Blue) versus non-couplings (Red). We additionally report the distances over the subset of `eaDCA`* couplings associated with good compatibility score ($PC > 0.8$).

Figure 7 Distance distribution of 4'-4' atomic distances obtained from PDB structures projected onto the RFAM reference alignments. Distances of more than 40Å are aggregated in the final bin.

4.3 Geometric distances of eaDCA couplings

We further estimate the ability of eaDCA to identify informative couplings towards 3D modeling, We consider tertiary structure data available in the RFAM alignments into our analysis. Among the 41 families selected previously, we were able to obtain high-quality 3D models, and alignments to their multiple sequence alignments within RFAM [20], for 6 of them (see Appendix Note A.2). We retrieved the corresponding 42 single chains from the PDB database [22], and base pair annotations produced using the DSSR [17] software suite, from the NAKD database [16]. For all 3D models, we computed the geometric distance between all pairs of 4' atoms (backbone) in the alignment, and report in Figure 7 the distances stratified by presence/absence in the consensus and inferred eaDCA model.

Figure 7a mainly represents a validation of the alignments of 3D models to RFAM family. As expected, for base pairs in the RFAM consensus, the distribution of 4'-4' distance is extremely concentrated around 6.5 Å, in agreement with textbook knowledge in structural biology. The distance distribution for consecutive nucleotides is more widespread, but consistent with the expected distance between nucleotides separated by at most 3 nucleotides. Finally, pairs of columns, associated with non-canonical base pairs in at least one aligned PDB structure, induce 4'-4' distances that are similar on average as consensus pairs, with higher yet reasonable variance. All of those distributions strongly differ from that of other pairs, which virtually always induce distances of more than 40Å. This confirms the quality of the structural alignments found in RFAM, and their potential utility in assessing couplings.

The distance distribution induced by couplings is more informative, and provides promising directions towards future predictive methods. As shown in Figures 7b and 8, the 3D 4'-4' distance induced by `eaDCA`*-produced couplings strongly differs from the non-couplings (background) distribution. The distance distribution for couplings is largely concentrated around the value of 6.5Å, as is expected for base pairs. Couplings also feature a minority of larger distances, inconsistent with direct base pairing, with some associated with distances greater than 40Å.

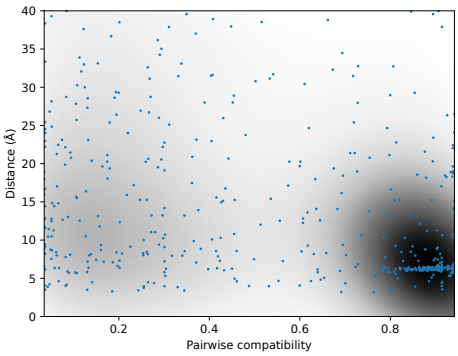


Figure 8 2D scatter plot of the distribution of coupling distances and their pairwise compatibility superimposed with an estimate of the density. Couplings with a distance above 40Å are not shown.

Those more distant couplings may indicate alignment/annotation errors, or reflect negative evolutionary pressure, which does not necessarily involve positions in direct contact. The hypothesis of a negative selective pressure is also partly supported by the fact that restricting our analysis to compatible couplings ($PC > 0.8$) greatly increases the population of the *canonical* 6.5Å bin, while more than halving the proportion of couplings at distances beyond 40Å.

5 Conclusion

In the context of the statistical modeling of RNA families, we revised the algorithmic framework underlying eaDCA [3], a proven greedy approach with the advantage of producing sparse and interpretable models. We exploited their sparsity using the stronger notion of treewidth, to devise rigorous, stable, deterministic and fixed-parameter tractable alternative algorithms for the modeling tasks (learning, sampling). We further evaluated the ability of our implementation, dubbed `eaDCA`*, to infer couplings with structurally significant roles. We did so in a systematic manner, and when applicable, using the available 3D structures. Our analysis shows that Direct-Coupling Analysis can be used to mine co-evolutionary signals present in RNA families, and that it is able to recover couplings lists associated with low tree widths, where the base pairs of the consensus structure feature prominently. We further verify that `eaDCA`* couplings are associated with low distances in PDB structures, and that for some of them that may show signs of negative evolutionary pressures. Our work suggests further investigations over the use and impact of non-greedy approaches for the choice of couplings (e.g. maximum coupling, Nussinov, local search), or the comparison with other standard approaches that make use of co-evolution [11].

References

1 Stefan Arnborg, Derek G Corneil, and Andrzej Proskurowski. Complexity of finding embeddings in ak-tree. *SIAM Journal on Algebraic Discrete Methods*, 8(2):277–284, 1987.

2 Hans L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '93, pages 226–234, New York, NY, USA, 1993. Association for Computing Machinery. `doi:10.1145/167088.167161`.

3 Francesco Calvanese, Camille N Lambert, Philippe Nghe, Francesco Zamponi, and Martin Weigt. Towards parsimonious generative modeling of RNA families. *Nucleic Acids Research*, 52(10):5465–5477, April 2024. doi:10.1093/nar/gkae289.

4 Eleonora De Leonardis, Benjamin Lutz, Sebastian Ratz, Simona Cocco, Rémi Monasson, Alexander Schug, and Martin Weigt. Direct-coupling analysis of nucleotide coevolution facilitates RNA secondary and tertiary structure prediction. *Nucleic Acids Research*, page gkv932, September 2015. doi:10.1093/nar/gkv932.

5 Sean R. Eddy. Homology searches for structural RNAs: from proof of principle to practical use. *RNA*, 21(4):605–607, March 2015. doi:10.1261/rna.050484.115.

6 Samuel Gardelle. eaDCAstar. Software, version 1., swhId: swh:1:dir:e53ad07a87e78c5d6e48b33986c10a1c937ed706 (visited on 2026-08-14). URL: https://codeberg.org/samuel-gardelle/wabi26-eaDCAstar, doi:10.4230/artifacts.27542.

7 Andreas Gruber, Ivo Hofacker, Stephan Bernhart, Peter Stadler, and Sebastian Will. RNAalifold: improved consensus structure prediction for RNA alignments. *BMC Bioinformatics*, 9(Art. 474), 2008. doi:10.1186/1471-2105-9-474.

8 Paul G. Higgs and Niles Lehman. The RNA world: molecular cooperation at the origins of life. *Nature Reviews Genetics*, 16(1):7–17, November 2014. doi:10.1038/nrg3841.

9 Mark Jerrum and Alistair Sinclair. Polynomial-time approximation algorithms for the Ising model. *SIAM Journal on Computing*, 22(5):1087–1116, 1993. doi:10.1137/0222066.

10 Ioanna Kalvari, Joanna Argasinska, Natalia Quinones-Olvera, Eric P Nawrocki, Elena Rivas, Sean R Eddy, Alex Bateman, Robert D Finn, and Anton I Petrov. Rfam 13.0: shifting to a genome-centric resource for non-coding RNA families. *Nucleic acids research*, 46:D335–D342, January 2018. doi:10.1093/nar/gkx1038.

11 Aayush Karan and Elena Rivas. All-at-once RNA folding with 3d motif prediction framed by evolutionary information. *Nature Methods*, 22(10):2094–2106, 2025.

12 Kenneth Katz, Oleg Shutov, Richard Lapoint, Michael Kimelman, Rodney Brister, and Christopher O'Sullivan. The sequence read archive: a decade more of explosive growth. *Nucleic Acids Research*, 50(D1):D387–D390, November 2021. doi:10.1093/nar/gkab1053.

13 Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

14 Camille N Lambert, Vaitea Opuu, Francesco Calvanese, Polina Pavlinova, Francesco Zamponi, Eric J Hayden, Martin Weigt, Matteo Smerlak, and Philippe Nghe. Exploring the space of self-reproducing ribozymes using generative models. *Nature communications*, 16(1):7836, 2025.

15 Marlen C. Lauffer, Willeke van Roon-Mom, and Annemieke Aartsma-Rus. Possibilities and limitations of antisense oligonucleotide therapies for the treatment of monogenic disorders. *Communications Medicine*, 4(1), January 2024. doi:10.1038/s43856-023-00419-1.

16 Catherine L Lawson, Helen M Berman, Li Chen, Brinda Vallat, and Craig L Zirbel. The nucleic acid knowledgebase: a new portal for 3d structural information about nucleic acids. *Nucleic Acids Research*, 52(D1):D245–D254, January 2024. doi:10.1093/nar/gkad957.

17 Xiang-Jun Lu, Harmen J Bussemaker, and Wilma K Olson. DSSR: an integrated software tool for dissecting the spatial structure of RNA. *Nucleic acids research*, 43(21):e142–e142, 2015.

18 Mihir Metkar, Christopher S. Pepin, and Melissa J. Moore. Tailor made: the art of therapeutic mRNA design. *Nature Reviews Drug Discovery*, 23(1):67–83, November 2023. doi:10.1038/s41573-023-00827-x.

19 Faruck Morcos, Andrea Pagnani, Bryan Lunt, Arianna Bertolino, Debora S. Marks, Chris Sander, Riccardo Zecchina, José N. Onuchic, Terence Hwa, and Martin Weigt. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–E1301, 2011. doi:10.1073/pnas.1111471108.

20 Nancy Ontiveros-Palacios, Emma Cooke, Eric P Nawrocki, Sandra Triebel, Manja Marz, Elena Rivas, Sam Griffiths-Jones, Anton I Petrov, Alex Bateman, and Blake Sweeney. Rfam 15: RNA families database in 2025. *Nucleic Acids Research*, 53(D1):D258–D267, January 2025. doi:10.1093/nar/gkae1023.

21 The ENCODE Project Consortium. Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. *Nature*, 447(7146):799–816, June 2007. doi:10.1038/nature05874.

22 wwPDB consortium. Protein data bank: the single global archive for 3d macromolecular structure data. *Nucleic Acids Research*, 47(D1):D520–D528, January 2019. doi:10.1093/nar/gky949.

23 Hua-Ting Yao, Jérôme Waldispühl, Yann Ponty, and Sebastian Will. Taming disruptive base pairs to reconcile positive and negative structural design of rna. In *RECOMB 2021-25th international conference on research in computational molecular biology*, 2021.

A Appendix

The proofs in this section rely on the following lemma:

▶ **Lemma 12** (Separation lemma). *Let $\mathcal{T}$ be a tree decomposition and $v \in V(T)$ one of its nodes. Let $C_1, .., C_k$ be the connected components obtained by removing v from T. Let $\mathcal{X}_l = \bigcup_{v \in C_l} X_v$ be the nodes of G involved in each component. Then we have that:*

$$X_v \sqcup \bigsqcup_{1 \leq l \leq k} \mathcal{X}_l - X_v = V(G)$$

Proof. Assume to the contrary that there exists $l_0, l_1, x \in V(G)$ such that $x \in (\mathcal{X}_{l_0} - X_v) \cap (\mathcal{X}_{l_1} - X_v)$ and $l_0 \neq l_1$. Let $v_0 \in C_{l_0}$ (resp. $v_1 \in C_{l_1}$) be such that $x \in X_{v_0}$ (resp. $x \in X_{v_1}$). Then since T is a tree there is a unique path from v_0 to v_1, with v on this path since it separated C_{l_0} and C_{l_1}. Because $x \in X_{v_0} \cap X_{v_1}$ by the running intersection property (see Definition 4) we have that $x \in X_v$ which is a contradiction with $x \in \mathcal{X}_{l_0} - X_v$. ◀

▶ **Lemma 6** (DP equation for the partition function). *Let $v \in V(T)$ be a vertex of the decomposition such that: $u_1, .., u_k$ are the direct children of v in T. Then the following holds:*

$$Z_v(s \in \Sigma^{X_v \cap X_{pa(v)}}) = \sum_{\substack{s' \in \Sigma^{X_v} \\ s \subset s'}} e^{-\sum_{J_{i,j} \in f^{-1}(v)} J_{i,j}(s'(i), s'(j))} \times \prod_{1 \leq i \leq k} Z_{u_i}(s'|_{X_{u_i} \cap X_v})$$

where $s|_S$ is the restriction of s onto the columns of S.

Proof. Let $C_0, C_1, .., C_k$ be the connected components obtained by removing v from T, such that:

- C_0 is the connected component containing the parent of v.
- $u_l \in C_l$ is the child of v adjacent to v.

This decomposition of T yields the following decomposition of the energy $E(w)$:

$$E(w) = \left(\sum_{1 \leq l \leq k} E_{v_l}(w) \right) + \left(\sum_{J_{i,j} \in f(v)} J_{i,j}(w_i, w_j) \right)$$

Moreover, by definition we have that $E_{v_l}(\cdot)$ only depends on $w_{|Y_{u_l}}$ since $Y_{u_l} = \mathcal{X}_l$ (u_l is the root of C_l):

$$E(w) = \left(\sum_{1 \leq l \leq k} E_{v_l}(w_{|Y_{u_l}}) \right) + \left(\sum_{J_{i,j} \in f(v)} J_{i,j}(w_i, w_j) \right)$$

Use this to rewrite the definition of the partition function $Z_v(s \in \Sigma^{X_v \cap X_{pa(v)}})$:

$$Z_v(s) = \sum_{\substack{s' \in \Sigma^{Y_v} \\ s \subset s'}} e^{-E(s')} = \sum_{\substack{s' \in \Sigma^{Y_v} \\ s \subset s'}} \left(e^{-\sum_{J_{i,j} \in f(v)} J_{i,j}(s'(i), s'(j))} \right) \prod_{1 \leq l \leq k} e^{-E_{u_l}(s'_{|Y_{u_l}})}$$

By applying Lemma 12, we have that the sets $(Y_{u_l} - X_v)_l$ are disjoint, which allows us to decompose the choice of s':

$$\begin{aligned} Z_v(s) &= \sum_{\substack{s' \in \Sigma^{X_v} \\ s \subset s'}} \left(e^{-\sum_{J_{i,j} \in f(v)} J_{i,j}(s'(i), s'(j))} \right) \prod_{1 \leq l \leq k} \sum_{\substack{s_l \in \Sigma^{Y_{u_l}} \\ s' \subset s_l}} e^{-E_{u_l}(s_l)} \\ &= \sum_{\substack{s' \in \Sigma^{X_v} \\ s \subset s'}} \left(e^{-\sum_{J_{i,j} \in f(v)} J_{i,j}(s'(i), s'(j))} \right) \prod_{1 \leq l \leq k} \mathcal{Z}_{u_l}(s'_{|X_{u_l} \cap X_v}) \end{aligned}$$

◀

▶ **Lemma 7.** *Let $v_1, .., v_n$ be a reversed topological ordering of T starting from the root. The following algorithm samples from the Boltzmann distribution of M:*

Let s be the unique assignment of $\Sigma^{\emptyset}$.
for $1 \leq i \leq n$ ***do***
Let $B = X_{v_i} - dom(s)$.
Sample $s' \in \Sigma^B$ with probability proportional to $Z_{v_i}((s + s')_{|X_{v_i} \cap X_{pa(v_i)}})$.
Extend s with the values of s'.
end for

Proof. To show the correctness of the algorithm, it suffices to show that at each step we have that:

$$\forall s' \in \Sigma^B : Z_{v_i}((s + s')_{|X_{v_i} \cap X_{pa(v_i)}}) \propto P(s'|s)$$

The ordering $v_1, .., v_n$ guarantees that when a node v_i is visited, none of its descendants has been visited yet. Moreover, letting u be the parent of v_i we can split T by removing u from it and v_i will be the root of the connected component composed of its descendants. Considering the other components together, this induces a partition of $V(T)$ as $D_{v_i} \sqcup A$ where A is the set of non-descendants of v_i. This induces the following decomposition of the energy function:

$$E(w) = E_{v_i}(w) + E_A(v_i) \text{ with } E_A(w) = \sum_{\substack{J_{i,j} \in u \\ u \in A}} J_{i,j}(w_i, w_j)$$

that can be used to rewrite $P(s'|s)$:

$$P(s'|s) \propto \sum_{\substack{w \in \Sigma^N \\ s, s' \subset w}} e^{-E(w)} = \sum_{\substack{w \in \Sigma^N \\ s, s' \subset w}} e^{-E_{v_i}(w)} \times e^{-E_A(w)}$$

Moreover, letting $\mathcal{X}_A = \cup_{u \in A} X_u$, the decomposition of $V(T)$, together with Lemma 12 implies that: $\mathcal{X}_A \cap Y_{v_i} \subset X_v$. Except for the columns of X_{v_i}, E_A and E_{v_i} involve different sets of columns, which allows us to further simplify:

$$P(s'|s) \propto \left(\sum_{\substack{a \in \Sigma^A \\ s \subset a}} e^{-E_A(a)} \right) \times \left(\sum_{b \in \Sigma^{Y_{v_i} - X_{v_i}}} e^{-E_{v_i}(s'+s+b)} \right)$$

$$= \underbrace{\left(\sum_{\substack{a \in \Sigma^A \\ s \subset a}} e^{-E_A(a)} \right)}_{\text{constant with respect to } s'} \times \mathcal{Z}_{v_i}((s+s')_{|X_{v_i} \cap X_{pa(v_i)}}) \quad (\star)$$ ◀

▶ **Lemma 10.** *Let $v \in V(T)$ and P_v be the marginal of the Boltzmann distribution restricted to the columns of $X_v \cap X_{pa(v)}$. Then for $s \in \Sigma^{X_v \cap X_{pa(v)}}$ we have that:*

$$Z_v^o(s) \times Z_v(s) = \sum_{\substack{s' \in \Sigma^{\{1,..,N\}} \\ s \subset s'}} e^{-\beta E(s')} \propto P_v(s)$$

Proof. If v has no parent then the proof is direct: $Z_v^o(\emptyset) = 1$ and $Z_v(\emptyset) = \mathcal{Z}$. Otherwise suppose u is the parent of v. Then consider the proof of Lemma 7 : split the decomposition $\mathcal{T}$ by removing u from T and assembling the non-descendants of v together. Now observe that the energy function E_A in the proof corresponds in fact to E_v^o. Then note that equation $(\star)$ directly gives the result that we seek. ◀

▶ **Lemma 11** (Augmented tree decompositions)**.** *Let $\mathcal{T}$ be a tree decomposition of G of size $s \in \mathbb{N}$ and $x \in V(G)$. We can build a decomposition $\mathcal{T}^x$ of size $2s$ and treewidth at most $tw(\mathcal{T}) + 1$ such that for every $y \in V(G)$ there exists $v_y \in V(T^x)$ such that:*

$$\{x, y\} \subset X_{v_y} \cap X_{pa(v_y)}$$

Proof. Construct $\mathcal{T}^x$ from $\mathcal{T}$ as follows:

- Add x to every bag of $\mathcal{T}$ if it is not already present.
- Add an additional child v' to every node v such that $X_{v'} = X_v$.

The size of bags is increased by at most 1 which justifies the bound on the treewidth of $\mathcal{T}^x$. We can verify that all the axioms of tree decompositions are verified. The first two axioms are inherited from $\mathcal{T}$, while the third is classically [13] equivalent to:

For $z \in V(G)$ the nodes of T whose bag contain z form a connected subtree of T.

Because the duplication of bags is adjacent it doesn't create further violations of this axiom, and moreover for $z = x$ the subtree is the whole tree, thus necessarily connected. It remains to be shown that $\{x, y\} \subset X_{v_y} \cap X_{pa(v_y)}$ for some $v_y \in V(T^x)$. By the first axiom of tree decompositions, there exists $v_0 \in V(T)$ such that $y \in X_{v_0}$. Take v_1 its correspondence in $V(T^x)$ and $v_y = v_1'$ the child of v_1 with the same bag. Then we have: $y \in (\{x\} \cup X_{v_0}) = X_{v_1} = X_{v_1'}$ and $x \in X_{v_1'} = X_{v_1}$ by definition. Thus $\{x, y\} \subset X_{v_y} \cap X_{pa(x_y)}$. ◀

A.1 List of selected RNA families

Collection of 41 RFAM families considered in our case study:

Family	N	n	PCC	$\|E\|$	tw	Family	N	n	PCC	$\|E\|$	tw
RF00001	230	712	0.76	148	5	RF00003	203	100	0.42	200	5
RF00005	118	954	0.83	129	6	RF00013	262	150	0.39	200	3
RF00019	145	104	0.59	159	6	RF00020	192	180	0.41	199	6
RF00050	221	146	0.37	200	3	RF00167	113	133	-0.059	128	6
RF00169	127	261	0.64	134	6	RF01051	136	155	0.23	144	6
RF01689	112	146	0.3	133	6	RF01695	129	456	0.68	135	6
RF01699	264	194	0.47	200	3	RF01731	193	210	0.61	189	6
RF01745	345	165	0.38	200	1	RF01831	179	101	-0.026	193	6
RF01852	119	109	0.53	136	6	RF02034	248	172	0.55	200	4
RF02035	153	320	0.78	152	6	RF02678	228	155	0.52	200	3
RF02921	200	143	0.52	200	5	RF02928	130	236	0.53	124	6
RF02932	163	102	-0.03	169	6	RF02944	104	224	0.68	123	6
RF02957	232	448	0.86	192	6	RF02960	240	155	0.54	200	4
RF02966	173	108	0.16	179	6	RF02967	154	139	0.37	159	6
RF02968	114	229	0.53	128	6	RF02972	128	154	0.78	139	6
RF02976	155	210	0.79	156	6	RF02987	121	481	0.72	129	6
RF02996	118	167	0.33	132	6	RF02999	102	248	0.69	103	6
RF03000	203	368	0.5	200	5	RF03005	201	129	0.39	170	6
RF03021	119	141	0.59	134	6	RF03074	117	261	0.51	120	6
RF03075	115	209	0.72	130	6	RF03127	110	370	0.59	125	6
RF03147	112	149	0.54	129	6						

N: alignment width, n: alignment height, PCC: pearson correlation between covariations of alignment and model (as in [3]), $|E|$: number of activated couplings, tw: final treewidth

A.2 Processing of geometric data

The alignments provided by RFAM contain occasionally sequences annotated with PDB structures that we were able to access. These structures enable us to reconstruct distances between residues in the family. Similarly, annotation tools such as DSSR were used to reconstruct non-canonical basepairs using these tertiary structures. The distances and annotations are computed for positions relative to the real ungapped RNA sequence present in the PDB. It is then only a matter of projecting this sequence into the alignment's column space to be able to compute distances/annotations in the reference of the alignment. However, due to what we believe to be frequent annotations errors of the PDB chains, we had to perform pairwise alignments between the chains of residues actually present in the PDBs, with the sequence reported in the family alignment. Some structures were rejected when this pairwise alignment was deemed of poor quality, this explains why some families with tertiary structures available were not included. Moreover, since the structures correspond to sequences that may contain deletions, in the reference of a family's alignment, not all positions are represented in 3D structures, and some may be represented many times in different structures. When a pair of position is present in multiple structures, we choose to use the minimum distance reported, and when distances were not available for pairs of positions corresponding with a coupling, it was excluded from the analyses requiring distance information.

A.3 Evolution of the treewidth

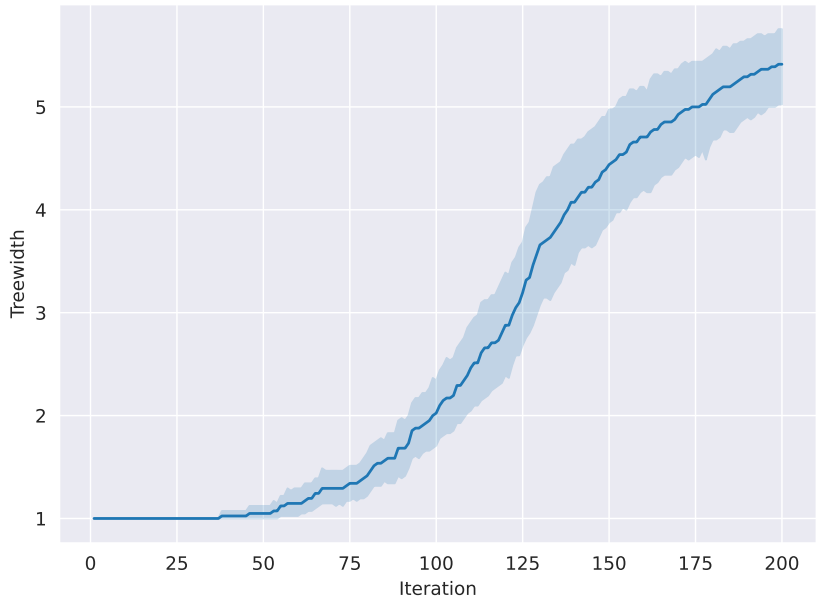


Figure 9 Increase in average treewidth (and 95% percentiles) at different iterations. Instances reaching treewidth 6 are no longer iterated upon but are kept in the distribution.