Old dog, new tricks: Exact seeding strategy improves RNA design performances

Théo Boury, Leonhard Sidl, Ivo L. Hofacker, Yann Ponty, Hua-Ting Yao

▶ **To cite this version:**

Théo Boury, Leonhard Sidl, Ivo L. Hofacker, Yann Ponty, Hua-Ting Yao. Old dog, new tricks: Exact seeding strategy improves RNA design performances. 2024. hal-04756160

HAL Id: hal-04756160
https://hal.science/hal-04756160v1

Preprint submitted on 28 Oct 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Old dog, new tricks: Exact seeding strategy improves RNA design performances

Théo Boury[1 [0009−0004−0553−4789]], Leonhard Sidl[2,3 [0009−0006−6440−4807]], Ivo L. Hofacker[2,3 [0000−0001−7132−0800]], Yann Ponty[1 [0000−0002−7615−3930]], and Hua-Ting Yao[2 [0000−0002−1720−5737]]

[1] Laboratoire d'Informatique de l'Ecole Polytechnique (LIX; UMR 7161), Institut Polytechnique de Paris, France
{theo.boury,yann.ponty}@lix.polytechnique.fr
[2] Department of Theoretical Chemistry, University of Vienna, 1090 Vienna, Austria
{sidl,ivo,htyao}@tbi.univie.ac.at
[3] Faculty of Computer Science, Research Group Bioinformatics and Computational Biology, University of Vienna, 1090 Vienna, Austria

Abstract. The Inverse Folding problem involves identifying RNA sequences that adopt a target structure with respect to free-energy minimization, i.e. preferential to all alternative structures. The problem has historically been regarded as challenging, largely due to its proven NP-completeness of an extended version where the base pair maximization energy model is used. In contrast, it has recently been shown that a large subset called m-separable structures, notably including those comprising helices of length 3+, can be solved in linear-time within the same energy model. This permits not only the identification of a single solution, but also the characterization of a language of solutions.
In this work, we seek to describe the "hardness" of Inverse Folding, bridging (at least heuristically) the gap between a simplified energy model and a more realistic Turner energy model. We used `LinearBPDesign` to generate seed sequences for `RNAinverse`, thereby improving the design process in a Turner energy model. To this end, we extended `LinearBPDesign` to accommodate biseparability and to handle non- or high modulo separable structures by minimalist addition of base pairs.
Our study suggests that seeds generated by `LinearBPDesign` capture long-range interactions, thereby improving the performance of `RNAinverse` compared to seed focusing on refining the energy model itself. Most surprisingly, a significant number of `LinearBPDesign` seeds uniquely fold into the target structure in the Turner model, especially when helices are at least of length 2. This observation suggests that the "hardness" of design may arise from the intrinsic properties of the structures themselves.

Keywords: RNA design · RNA secondary structure · Dynamic programming · Sampling.

1 Introduction

A recurrent problem in RNA structural design, called inverse folding, consists in finding RNA sequences that preferentially fold into one (or several) user-provided structures. Targeting a certain structure is indeed an objective of interest as the structure of biologically-active non-coding RNAs is often seen as an important contributor to its function [4]. Considering the wealth of biological functions (catalytic, regulatory...) performed by RNA, including but not limited to gene expression, splicing and epigenetic modifications [19], rational design of synthetic and diverse RNA appears more and more of high importance in order to unlock applications in synthetic biology and medicine [9,5,11,26].

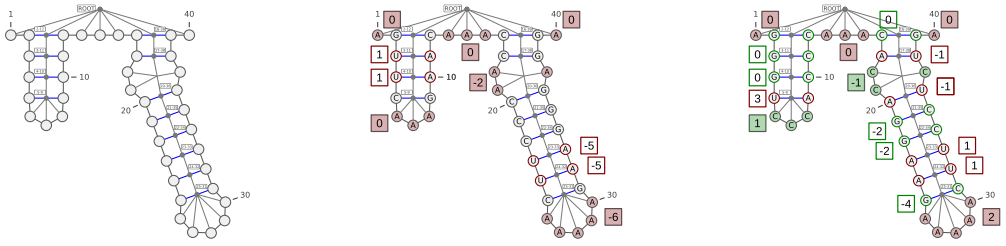
The first method proposed for inverse folding, named `RNAinverse` [8], was developed in 1994 and explores the sequence space by applying mutations to an initial sequence. This heuristic strategy has been iterated on and improved numerous times over the last years with notable examples including `RNA-SSD` [1] and `FRNAkenstein` [12]. Newer methods such as `INCARNAFBINV` [16] or `RNAPOND` [25] combine the negative and positive design paradigm to find an optimal solution. `NEMO` [14] integrates domain knowledge into a Nested Monte Carlo Search in order to achieve results similar to those of expert human designers. With its score of 94 out of 100, `NEMO` currently outperforms all other design tools on the `EterRNA` 100 benchmark [10]. Finally, solutions relying on machine learning have been increasingly developed in recent years. `SentRNA` [21] employs a fully connected neural network, trained on player-submitted solutions to the `EterRNA` game in combination with an adaptive walk to further refine the results. In comparison, `libLEARNA` [18] utilizes automated deep reinforcement learning to train a policy network.

An increased level of attention has been dedicated to the theoretical and computational properties of RNA inverse folding. Due to the intricacies of the Turner nearest-neighbor free-energy model [22], very little is currently known about inverse folding in realistic energy models. Yao *et al.* [23] characterized an exhaustive set of local undesignable motifs through brute force enumeration, and revealed a drastic reduction of the set of designable secondary structures. Zhou *et al.* [28] extended the collection by larger undesignable structures, instances of which were detected in a popular RNA inverse benchmarks [10]. Inverse folding under a simplified BP energy models (*aka BP inverse folding*), in which independent additive contributions of individual base pairs are assumed, enjoy more comprehensive theoretical studies. Bonnet *et al.* [2] established the NP-hardness of a mildly constrained version of inverse folding in a BP energy model. A tree-coloring perspective introduced by Hales *et al.* [6] led to a characterization of easy classes of instances for inverse folding. Surprisingly, this framework was instantiated by Boury *et al.* [3] into a linear-time solution for all secondary structures consisting of helices having 3^+ BPs. The underlying DP algorithm could be extended into a uniform random generation of solutions for BP inverse folding, and a sizeable portion of solutions were shown to represent promising solutions for inverse foldifng in the Turner energy model.

The existence of a linear-time algorithm for sampling sequences, guaranteed to be solutions to inverse folding in the BP model, motivates consideration of exact BP designs as seeds within classic local optimization schemes. We focus on `RNAinverse` due to both its historical value, relatively straightforward optimization scheme, and surprising resilience in the context of a large array of competitors. Firstly, we wish to assess the performance of the exact design, in BP energy models, as a candidate solution in the Turner model. Of particular interest is the number of single-point mutations needed to convert an exact BP design into a solution/design with respect to the Turner model. Finally, a realistic design scenario requires the generation of (large) sets of diverse solutions, to capture further constraints through *post hoc* filtering, motivating the consideration of generalized strategies for sequence generation.

Our main contributions and conclusions include:

1. Exact design in simplified models systematically outperforms naive seeding strategies, leading to ultimate solutions that are more stable and substantially diverse (see Sec. 3.1 and 3.2);
2. The introduction of the concept of biseparability (see Sec. 2.2), strictly generalizing the concept of separability [3] to increase the diversity of produced solutions (see Sec. 3.2);
3. A uniform random generation algorithm for biseparated sequences (see Sec. 2.2), guaranteed to represent solutions to BP inverse folding, running in linear-time for a large class of structures, notably including secondary structures consisting of helices having 3^+ BPs;
4. We observe that an excessive focus on a limited collection of benchmarks, consisting in part of pathological structures, results in the promotion of method that poorly generalize (see Sec. 3.4). More extensive validation efforts are needed to ensure the continued development of general methodologies.



Target secondary structure | 2-separated sequence | (2, 2)-biseparated sequence

Fig. 1: **Secondary structure as a tree and levels associated with (bi)separated sequences.** (Left) The tree represents base pairs as (internal) nodes, and unpaired positions as leaves. (Center) Example of a proper 2-separated sequence ($\Rightarrow \mathcal{B}$ design). The levels of unpaired (A) positions are even, while the levels of paired AU/UA positions are odd, implying 2-separability. (Right) Example of a proper $(2, 2)$-biseparated sequence ($\Rightarrow \mathcal{B}$ design). Two sets of levels (red for AU/UA, and green for GC/CG) are simultaneously handled relatively to A and C, and the absence of overlap ensures unicity of the MFE fold, while allowing the assignment of a mix of A and C to unpaired regions.

2 Methods

In this work we consider the Inverse Folding problem, the most typical instance of negative RNA design.

Definition 1 (Inverse Folding).
Input: *A nested secondary structure S of length n.*
Output: *RNA sequence ω with $|\omega| = n$, such that $\forall S' \neq S, \mathcal{E}(\omega, S) < \mathcal{E}(\omega, S')$ where $\mathcal{E}(\omega, S)$ is the free-energy of ω folding into S within some energy model of interest.*

In other words, the Inverse Folding problem not only requires the target structure to be the minimum free-energy (MFE) conformation of ω, but also requires the absence of competing folds of equal stability. It is only in that case that we say that ω is a design for S. In this work, we study Inverse Folding in two different energy models:

1. The base pair maximization energy model $\mathcal{B}$, sometimes referred to as the Nussinov-Jacobson energy model [13], where the energy of a structure is simply defined as minus its number of base pairs;
2. The – more realistic – Turner energy model $\mathcal{T}$, where the energy of a secondary structure is defined as a sum of independent contributions associated with loops occurring in the structure. A precise definition of loops is not needed here, and thus omitted in the interest of space, but we refer the reader to Turner and Mathews [22] for details.

To lift any ambiguity, we say a structure S is $\mathcal{B}$-*designable* (resp. $\mathcal{T}$-*designable*) if there exists a sequence ω which is a $\mathcal{B}$-design (resp. $\mathcal{T}$-design), *i.e.* a solution of inverse folding with respect to the model $\mathcal{B}$ (resp. $\mathcal{T}$).

2.1 Classic separability and linear-time design within a BP energy model

We first recall key concepts underpinning the `LinearBPDesign` method, with a focus on m-separability [3], reframed in terms of nucleotide assignments instead of the colors used by Hales *et al* [6]. We abstract a (target) *secondary structure* of length n as a tree $T = (V, E)$ (Fig. 1 left) where each node either represents a base pair (i, j) (internal nodes) or an unpaired position k (leaves). A *loop* consists in the union of an internal node (*i.e.* a base pair) and its (direct) children. For a sequence ω, the *content $C_{i,j}$ of a loop*, rooted at a base pair (i, j), is defined as the list of base pairs assigned to the children of (i, j), augmented with the inverted content of (i, j), *i.e.* $C_{i,j} := [\omega_j.\omega_i] \cdot [\omega_{i'}.\omega_{j'} \mid (i', j')$ BP children of $(i, j)]$.

Definition 2 (Proper sequence). *A sequence ω is proper for a secondary structure $T = (V, E)$ when, for each node $(i, j) \in V$, the content $\Psi := C_{i,j}$ of the loop rooted at (i, j) obeys:*

$$|\Psi|_{GC} \leq 1, |\Psi|_{CG} \leq 1, |\Psi|_{CG} \cdot |\Psi|_{GC} = 0, |\Psi|_{AU} \leq 1, |\Psi|_{UA} \leq 1 \text{ and } |\Psi|_{AU} \cdot |\Psi|_{UA} = 0,$$

where $|\Psi|_{XY}$ denotes the number of occurrences of XY in the list Ψ.

In other words, a sequence is *proper* when its loop assignments forbid local alternatives (Fig. 3). Being proper is thus a necessary condition for a sequence to represent a $\mathcal{B}$-design. Meanwhile, the *(modulo) m-separated* condition represents a sufficient condition to rule out the existence of global alternatives (*i.e.* long-range rearrangment) to the target T. It crucially relies on the concept of *level* which ensures G/C imbalance, and thus a strict suboptimality of alternatives, upon forming an alternative base pair.

Namely, given a sequence ω compatible with a structure T, the *level* $L : V(T) \to \mathbb{Z}$ of a node v is $L(v) := |p_v|_{\mathsf{GC}} - |p_v|_{\mathsf{CG}}$ where p_v denotes the base pairs found on the path from parent(v) to the root of T. Denote by $\mathcal{L}_{\mathsf{AU|UA}}(T;\omega)$ the set of levels of $\{\mathsf{AU}, \mathsf{UA}\}$ base pairs, and by $\mathcal{L}_{\mathsf{A}}(T;\omega)$ those of A-assigned unpaired positions.

Definition 3 ((Modulo) m-separated sequence). *A sequence ω is m-separated for a structure T, if and only if ω is proper, features A unpaired positions, and $\{l \bmod m \mid l \in \mathcal{L}_{\mathsf{AU|UA}}(T;\omega)\} \cap \{l' \bmod m \mid l' \in \mathcal{L}_{\mathsf{A}}(T;\omega)\} = \varnothing$.*

A structure T is *m-separable* if it admits an m-separated sequence. An example of a 2-separated sequence can be seen in Figure 1.

Finally, Boury *et al* [3] shows that it is sufficient for a sequence to be m-separated to represent a $\mathcal{B}$-design, *i.e.* to be a solution to Inverse Folding with respect to the $\mathcal{B}$ energy model thus avoiding all alternative structures made of AU, GC and also GU base pairs. Moreover, finding a proper m-separated sequence for a structure over n nucleotides (if it exists), can solved in $\mathcal{O}(n\,m\,2^m)$ time. The authors finally show that any designable structure featuring helices of length at least 3 is 2-separable, thus the Inverse Folding in $\mathcal{B}$ can be solved in linear-time for this large subset of reasonable instances.

2.2 Biseparated sequences: enriching the set of exact solutions for the BP-based model

An obvious limitation of m-separated sequences as defined above is that their unpaired positions are always set to A, yielding uncanny sequences of limited diversity. To work around the issue, we introduce the concept of biseparability, whereby both A and C are allowed in the unpaired regions. In addition to the G/C imbalance exploited by classic separability, biseparability captures A/U imbalance. It ensures that alternative G/C base pairs remain suboptimal when involving the C nucleotides intended to remain unpaired in T. Figure 1 shows an example of a (2,2)-biseparated sequence.

Concretely, we will now consider two types of levels: the *A-level*, denoted by $L_{\mathsf{A}} : V \to \mathbb{Z}$, refers to the classic level $L(v)$ introduced by Hales *et al* [6] and featured in the previous section; The *C-level* $L_{\mathsf{C}}(v)$ of a node T is similarly defined as:

$$L_{\mathsf{C}}(v) := |p|_{\mathsf{AU}} - |p|_{\mathsf{UA}} \quad (\text{and } L_{\mathsf{A}}(v) := L(v))$$

where p_v again denotes the base pairs found on the path from parent(v) to the root of T. For a given sequence ω, we denote by $\mathcal{L}_{\mathsf{CG|GC}}(T;\omega)$ the set of C-levels of $\{\mathsf{CG}, \mathsf{GC}\}$ base pairs, and by $\mathcal{L}_{\mathsf{C}}(T;\omega)$ those of C-assigned unpaired positions. These definitions enable the introduction of the concept of *(modulo)* $(m_{\mathsf{A}}, m_{\mathsf{C}})$*-biseparability* relative to A and C.

Definition 4 ((Modulo) (m_{A},m_{C})-biseparated sequences). *A sequence ω is* (m_{A},m_{C})-biseparated *for a target secondary structure T, if an only if:*

1. *ω is proper;*
2. *Levels of AU/UA and A do not overlap: $\{l \bmod m_{\mathsf{A}} \mid l \in \mathcal{L}_{\mathsf{AU|UA}}(T;\omega)\} \cap \{l' \bmod m_{\mathsf{A}} \mid l' \in \mathcal{L}_{\mathsf{A}}(T;\omega)\} = \varnothing$;*
3. *Levels of CG/GC and C do not overlap: $\{l \bmod m_{\mathsf{C}} \mid l \in \mathcal{L}_{\mathsf{CG|GC}}(T;\omega)\} \cap \{l' \bmod m_{\mathsf{C}} \mid l' \in \mathcal{L}_{\mathsf{C}}(T;\omega)\} = \varnothing$.*

Deciding biseparability in general is NP-hard as $(2n, 1)$-biseparability corresponds exactly to separability, which was proven NP-hard [3]. Similarly, deciding the existence of a (m_{A},m_{C})-biseparated sequence remains NP-hard in general, as it coincides with biseparability in the worst case (by considering $m_{\mathsf{A}} = 2n$ and $m_{\mathsf{C}} = 2n$). Our intent here is thus to explore small modular values of m_{A} and m_{C}.

Fortunately, even moderate values of m_{A} and m_{C} already capture large subsets of structures. For instance, it can be observed that any designable structure T with helices of size 3 or more admits a (2,1)-biseparated

sequence. Indeed, T then admits a sequence ω which is proper ($\Rightarrow$ Cond. 1) and 2-separated ($\Rightarrow$ Cond. 2), and additionally features A in each of its unpaired positions. It follows that $\mathcal{L}_{\mathsf{C}}(T;\omega) = \varnothing$, implying the validity of Cond. 3, and we conclude that ω is a (2,1)-biseparated sequence.

Theorem 1. *Given a structure T, any sequence ω compatible with T and biseparated is a $\mathcal{B}$-design.*

The proof is similar to that of separability [6], with minor modifications (see Sec. A for details).

A striking feature of $(m_{\mathsf{A}}, m_{\mathsf{C}})$-biseparated sequences is that, for fixed values of m_{A} and m_{C}, they can be found and uniformly sampled in time only linear in n, the size of the target. The following dynamic programming scheme counts the set of $(m_{\mathsf{A}},m_{\mathsf{C}})$-biseparated sequences for a target structure T, given admissible modular levels $\xi_{L_{\mathsf{A}}} \subset [1, m_{\mathsf{A}}]$ and $\xi_{L_{\mathsf{C}}} \subset [1, m_{\mathsf{C}}]$ respectively for unpaired As and Cs:

$$\mathrm{p}^{(\xi_{L_{\mathsf{A}}},\xi_{L_{\mathsf{C}}})}_{v\to\mu,(\ell_{\mathsf{A}},\ell_{\mathsf{C}})} = \begin{cases} \mathbb{1}_{(l\in\xi_{L_{\mathsf{A}}})\wedge(\mu=\mathsf{A})} + \mathbb{1}_{(l\in\xi_{L_{\mathsf{C}}})\wedge(\mu=\mathsf{C})} & \text{if } v \text{ is leaf} \\ 0 & \text{if } \ell \in \xi_{L_{\mathsf{A}}} \text{ and } \mu \in \{\mathsf{AU},\mathsf{UA}\} \\ 0 & \text{if } \ell \in \xi_{L_{\mathsf{C}}} \text{ and } \mu \in \{\mathsf{GC},\mathsf{CG}\} \\ 1 & \text{if children}(v) = \varnothing \\ \displaystyle\sum_{\substack{\mu' \text{ proper assignment} \\ \text{children}(v)\to\Sigma^2\cup\{\mathsf{A},\mathsf{C}\}}} \prod_{v_i\in\text{children}(v)} \mathrm{p}^{(\xi_{L_{\mathsf{A}}},\xi_{L_{\mathsf{C}}})}_{v_i\to\mu'(v_i),(\ell'_{\mathsf{A}},\ell'_{\mathsf{C}})} & \text{otherwise.} \end{cases}$$

where $v = (i, j)$ and $v = i$ correspond to a node or leaf of T, with prior nucleotide(s) assignment denoted by μ, ℓ_{A} and ℓ_{C} are the current modular A and C levels, and ℓ'_{A} and ℓ'_{C} the updated modular levels of A and C following the choice of μ': $\ell'_{\mathsf{C}} := \ell_{\mathsf{C}} + \mathbb{1}_{\mu'(v')=\mathsf{AU}} - \mathbb{1}_{\mu'(v')=\mathsf{UA}} \bmod m_{\mathsf{A}}$ and symmetrically for ℓ'_{A}. The overall number of separated sequences is then ultimately found in $\mathrm{p}(\xi_{L_{\mathsf{A}}}, \xi_{L_{\mathsf{C}}}) := \mathrm{p}^{(\xi_{L_{\mathsf{A}}},\xi_{L_{\mathsf{C}}})}_{\text{Root}(T)\to\varepsilon,(0,0)}$.

The correctness of the dynamic programming scheme can be established through a straightforward adaptation of the proof of m-separated sequences [3]. Moreover, for a fixed $(\xi_{L_{\mathsf{A}}}, \xi_{L_{\mathsf{C}}})$ pair, it can be computed in complexity which is linear in n the number of nucleotides, m_{A} and m_{C} since: i) μ may only take 8 possible values (single or pair of nucleotides); ii) ℓ_{A} and ℓ_{C} respectively take values in $[0, m_{\mathsf{A}})$ and $[0, m_{\mathsf{C}})$; iii) In the $\mathcal{B}$ model, a target structure featuring a loop having >4 BPs does not admit a solution to inverse folding ($\Leftarrow$ Proper condition is necessary for the existence of designs); iv) Apart from the open chain, only one type of A or C is allowed in the unpaired positions of the loop, otherwise a conflict would arise with the content of one of the base pairs of the multiloop. It follows from (iii) and (iv) that the sum over all assignments can be computed in constant time. By iterating over the $2^{m_{\mathsf{A}}+m_{\mathsf{C}}}$ possible values of $(\xi_{L_{\mathsf{A}}}, \xi_{L_{\mathsf{C}}})$, and checking if $\mathrm{p}(\xi_{L_{\mathsf{A}}}, \xi_{L_{\mathsf{C}}}) \neq 0$, one can determine the existence of modulo (m_{A}, m_{C})-separated sequences.

Theorem 2. *The existence of $(m_{\mathsf{A}}, m_{\mathsf{C}})$-biseparated sequences can be decided in $\mathcal{O}(n\, m_{\mathsf{A}}\, m_{\mathsf{C}}\, 2^{m_{\mathsf{A}}+m_{\mathsf{C}}})$.*

Meanwhile, a uniform generation of x biseparated sequences can be done in expected time $\mathcal{O}(x\, n\, 2^{m_{\mathsf{A}}+m_{\mathsf{C}}})$ following a precomputation in $\mathcal{O}(n\, m_{\mathsf{A}}\, m_{\mathsf{C}}\, 2^{m_{\mathsf{A}}+m_{\mathsf{C}}})$, as described in Boury *et al* [3]. In a nutshell, one chooses a random level assignment $(\xi_{L_{\mathsf{A}}}, \xi_{L_{\mathsf{C}}})$ with probability proportional to its number of sequences, and a rejection step is used to correct for the compatibility of certain sequences with at most $\mathcal{O}(2^{m_{\mathsf{A}}+m_{\mathsf{C}}})$ level assignments.

2.3 Minimal augmentation of non-separable secondary structures

Certain secondary structures are known to be non-separable with respect to the $\mathcal{B}$ energy model, implying the absence of m-separated proper sequences for any value of m. For generality, we should be able to handle these structures, so we chose to minimally modify the input structure. Indeed, it was shown by Halès *et al.* [6] that any structure with suitable loop constraints can become 2-separable by adding at most one base pair per helix. We revisit this idea in a more general setting, augmenting the input structure by at most k base pairs, with $\leq H$ additions per helix. Doing that, it is clear that we are not solving the initial problem, but at least we can get (and sample) exact sequences from slightly augmented structures, that lie in a controlled "neighborhood" from the input structure. The value of k should be as minimal as possible. In our case, we mostly work with $H = 1$ adding at most 1 base pair per helix as structures with minimally extended helices are likely to be functionally equivalent to the input instance. Figure 2 illustrates the process of structure augmentation with $k = 1$ and $H = 1$.

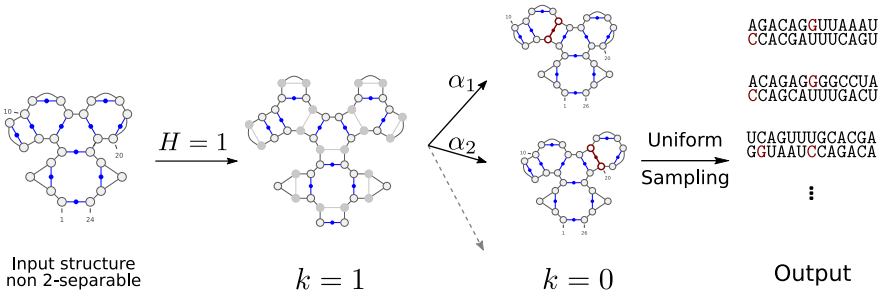


Fig. 2: **Non separable structure augmentation with $H = 1$ and $k = 1$.** At most $H = 1$ base pair (gray) could be added on each helix in the input structure. The value of k decreases by 1 whenever one base pair is added until the value reaches 0. The added base pair i (red) is selected with a probability proportional to the precomputed assignment counts α_i with dynamic programming. Then a sequence is sampled uniformly for the final augmented structure.

In terms of method, we simply tweak the dynamic programming. When assigned, each node v at the end of a helix can have a new behavior: it can do recursion on itself, artificially adding, at the bottom of v, a node v' that should also be assigned (thus giving v' a feasible proper assignment and incrementing the level as with v assigned). This auto-recursive call can be performed at most H times, effectively exploring the feasible augmented structures.

Note that for simplicity, the dynamic programming scheme is written for the m-separable case only, but can easily be adapted for the $(m_{\mathsf{A}}, m_{\mathsf{C}})$-biseparable case. Notations are the same as in Section 2.2 with k and H as defined above, and h represents the number of base pairs added on top of the "current" node:

$$\mathrm{p}^{\xi_L,H}_{v\to\mu,\ell,(k,h)} = \begin{cases} \mathbb{1}_{l\in\xi_L} & \text{if } v \text{ is leaf} \\ \mathrm{p}^{\xi_L,H}_{v\to\mu,\ell,(k,0)} + \sum_{\substack{\mu' \text{ proper assignment} \\ \text{for 1 child}}} \mathrm{p}^{\xi_L,H}_{v\to\mu'(v),\ell',(k-1,h+1)} & \text{if } k>0,\ h<H \text{ and } \mathrm{LH}(v) \\ 0 & \text{if } \ell\in\xi_L \text{ and } \mu\in\{\mathsf{AU},\mathsf{UA}\} \\ 1 & \text{if children}(v)=\varnothing \\ \sum_{\substack{\mu' \text{ proper assignment} \\ \text{children}(v)\to\Sigma^2\cup\{\emptyset\}}} \prod_{v_i\in\text{children}(v)} \prod_{\substack{\sum k_i=k \\ k_i=0 \text{ if } v_i \text{ is leaf}}} \mathrm{p}^{\xi_L,H}_{v_i\to\mu'(v_i),\ell',(k_i,0)} & \text{otherwise.} \end{cases}$$

where ℓ' corresponds to the next level after 1 assigned node as previously and LH is a function that returns True on input v iff v is a last node of a helix (*e.g.* has a leaf as a child or at least 2 base pairs children).

The complexity remains mainly the same as m-separability or $(m_{\mathsf{A}}, m_{\mathsf{C}})$-separability (Sec. 2) with minor overheads: i) $O(H)$ as every state can be duplicated H times in the worst case with $\leq H$ auto-recursive calls; ii) $\mathcal{O}(k^3)$ to distribute k among the ≤ 3 helices stemming from a multiloop. Since both of k and H remain limited to stay close to the intended target, the overall overhead remains generally inconsequential.

2.4 Heuristic extension for Turner energy model ($\mathcal{T}$-designability)

Multiloops. Even if large multiloops are forbidden in $\mathcal{B}$, structures in $\mathcal{T}$ can contain arbitrary large multiloops still remaining designable as the proper condition mainly forbids local rearrangement inside multiloops (Fig. 3). Thus we use the "unproper" strategy [3] that retains the condition on m-separability excluding the proper condition. The m-separability condition may be more useful in $\mathcal{T}$ as it catches "long-range interactions" by forbidding rerootings of any base pairs with all, even far, unpaired positions. Handling of multiloops of any size can be done without impacting performances, through a slight modification of the dynamic programming scheme. It involves auto-recursion as used in Section 2.3, with details found in Section F.

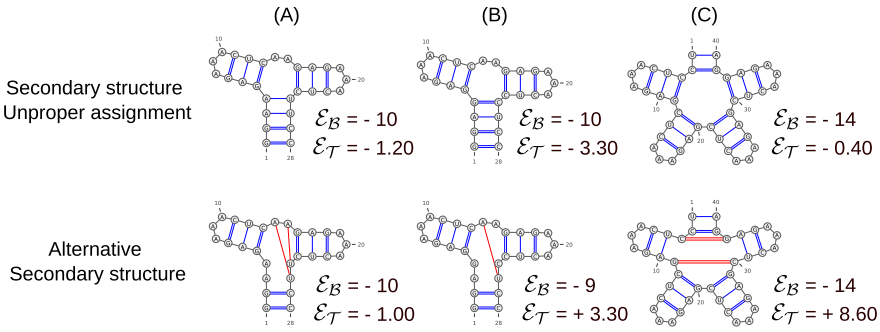


Fig. 3: **Structures containing forbidden motifs in $\mathcal{B}$ and alternatives over given (unproper) sequences** (A) The structure contains an undesignable motif (m3o). The Turner energy of the alternative is marginally worse than the target (-1.0 kcal.mol^{-1} vs -1.2 kcal.mol^{-1}). The sequence is not m-(bi)separated as AU and A occur at the same level; (B) For this sequence, the alternative is not competitive in $\mathcal{B}$. (C) The structure contains an m5 forbidden multiloop. Frequently, alternatives for the $\mathcal{B}$ model contain base pairs that are "isolated", and end up being uncompetitive in the $\mathcal{T}$ model.

Implementation and interfacing with `RNAinverse`. Given that a $\mathcal{B}$-design may not be a $\mathcal{T}$-design, we have combined the (bi)separated sequence generation (*i.e.* extended `LinearBPDesign`) with `RNAinverse` to design secondary structures in the Turner energy model. Given a target structure, a m-separated or $(m_\mathsf{A}, m_\mathsf{C})$-biseparated $\mathcal{B}$-design is uniformly sampled with $m, m_\mathsf{A}, m_\mathsf{C} \leq M$ and provided to `RNAinverse` as a starting seed sequence in the following three steps: i) The minimum modulo M is chosen such that an adequate number of $\mathcal{B}$-designs (1,000 by default) is included in the sampling pool. For biseparability, we enforce M to be at least 3 to ensure that there is a sufficient number of $\mathcal{B}$-designs with a mix of A and C in the unpaired region; ii) If a $\mathcal{B}$-undesignable multiloop presents, "unproper" strategy is used, and only in this case; iii) If M is "too large" or cannot be determined (non-separable structure), we use structure augmentation (Sec. 2.3) to decrease the modulo. However, in practice, this is rarely the case for a design in the $\mathcal{T}$model (see Sec. B).

From the seed sequence, `RNAinverse` performs an adaptive random walk in the sequence space. At each step, a position is randomly selected to be mutated as well as the paired partner (if one exists). The new sequence is accepted if the resulting base pair distance between the MFE structure and the target decreases. `RNAinverse` begins by targeting the small substructures and then progresses to the entire target structure to reduce the required computational time. Each walk stops when the target is reached, *i.e.* distance equals 0, or there is no more mutation that can be introduced to improve the distance. For the Inverse Folding problem, a negative flag should be used (option `-R-k`). `RNAinverse` restarts the walk from a new different seed sequence until k $\mathcal{T}$-designs are found. We slightly adjust `RNAinverse` for this study such that the target is the only MFE structure of the returned $\mathcal{T}$-design.

Other seed sequence generations. We also considered two other seed generation strategies to investigate the impact of `LinearBPDesign` exact seed on `RNAinverse`:

1. *Uniform sampling*: Default option of `RNAinverse`. A seed sequence is uniformly chosen from the entire sequence space. Each paired position takes nucleotides from six possible canonical base pairs;
2. *Boltzmann sampling*: Usually used in positive design. A seed sequence is sampled from a Boltzmann distribution based on the folding free-energy to the target [15]. We consider here a special base pair energy model from [7], where the energy contribution of each base pair depends on the nucleotides (prioritized to CG and GC) and on its position in the helix (stacked or helix-end).

We further limited all unpaired positions to be A for a better comparison with separated seeds.

3 Results

3.1 `LinearBPDesign` directly solves most structures with helices of length 2^+

The first query is whether $\mathcal{B}$-designs generated through `LinearBPDesign` are also $\mathcal{T}$-designs, particularly for structures comprising helix length of 3^+, given they are $\mathcal{B}$-designable. For minimum helix lengths of 1 and 2,

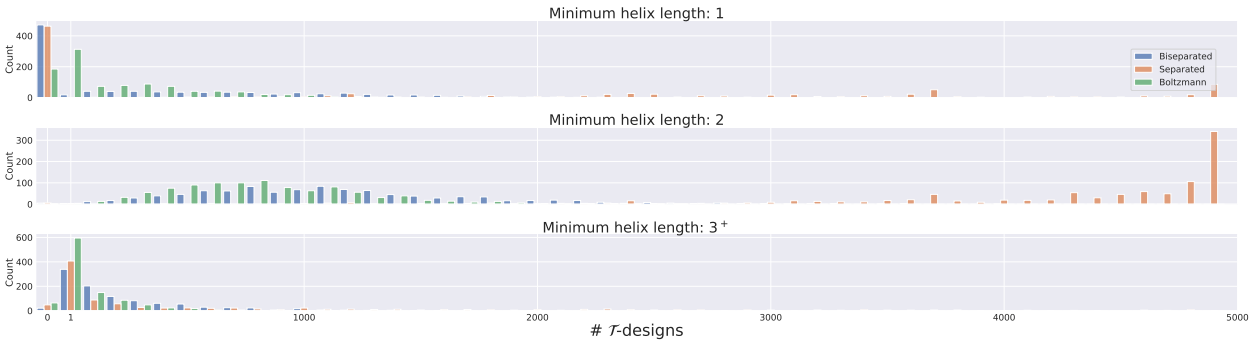


Fig. 4: **Histogram of structure count in the function of $\mathcal{T}$-design number.** From top to bottom, each set contains 1,000 structures of a minimum helix length of 1, 2, and 3^+ base pairs, respectively. For each structure, 5,000 sequences are sampled for each seed strategy, biseparability (blue), separability (orange), and Boltzmann sampling (green) without further `RNAinverse` optimization. At most one $\mathcal{T}$-design is found among 5,000 uniform sampled sequences for more than 2/3 of structures, the result is then omitted from the plot. The bin width is 100 except for the first bin, which represents the amount of unsolved structures.

the probability of a randomly and uniformly sampled structure being $\mathcal{T}$-designable decreases exponentially due to the existence of forbidden motifs with isolated base pairs or stacks [23]. To this end, three sets of 1,000 secondary structures of size 150 nts were created for different minimum helix lengths in two ways. The first set comprises 1,000 randomly and uniformly generated structures containing only helices of length 3^+. The second and the third sets are composed of each 1,000 MFE structures of random sequences computed with `RNAfold`, for minimum helix length of 1 and 2 base pairs respectively. Sequences with more than one MFE structure are excluded from the samples.

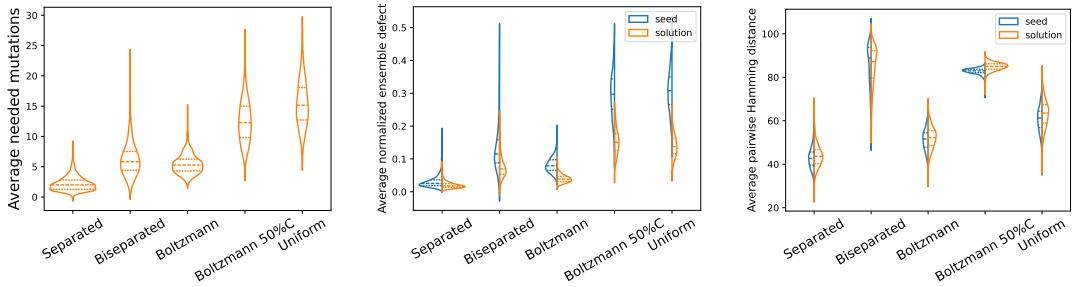
$\mathcal{B}$-designable structures are often $\mathcal{T}$-designable. For each structure, `LinearBPDesign` is employed with an appropriate minimum modulo M in order to uniformly sample 5,000 biseparated sequences. It is also possible that some sequences may be redundant as a result of the sampling process. However, the proportion of duplicates is relatively low, with an average of 0.423 duplicates per structure. As illustrated in Figure 4, at least one $\mathcal{T}$-design is present among the 5,000 sampled biseparated $\mathcal{B}$-designs for more than 95.8% of secondary structures containing only helices of length 2^+. Conversely, 421 out of 1,000 structures with an isolated base pair remain unsolved.

Separated $\mathcal{B}$-designs are mostly $\mathcal{T}$-designs for MFE structures. Nevertheless, only 20% of the identified $\mathcal{T}$-designs contain C in at least one unpaired position. Indeed, focusing exclusively on separated $\mathcal{B}$-designs markedly increases the number of $\mathcal{T}$-designs for the MFE structures. It is noteworthy that for 601 MFE structures with a minimum helix length of 2, over 90% of sampled separated $\mathcal{B}$-designs are also $\mathcal{T}$-designs. The discrepancy between these results and those on uniformly sampled structures indicates that the language employed in `LinearBPDesign` $\mathcal{B}$-designs may be surprisingly capable of capturing the characteristics inherent to the Turner energy model.

(Bi)Separability overlooks dangling energy. We examined the sequences sampled from a Boltzmann distribution, with the unpaired positions restricted to A. The outcomes are comparable to those of biseparated sequences, exhibiting, in general, a diminished degree of success, except for the structures containing isolated base pairs. Only for half of the structures with isolated base pairs, compared with the result of biseparability, no $\mathcal{T}$-design within 5,000 Boltzmann sampled sequences could be found. Two potential explanations for this phenomenon can be postulated. Firstly, the restriction on minimum (small) modulo-separated sequence space is too restrictive for these structures. Secondly, the simple Nussinov-Jacobson energy model fails to account for the energy contribution from dangling ends, while this is captured in the special base pair energy model used for Boltzmann sampling.

3.2 Negative seeds are typically close to solutions to the inverse folding problem

In this section, we aim to design within the Turner energy model the MFE secondary structures that cannot be directly solved by biseparated $\mathcal{B}$-designs in the previous section. It consists of in total 476 instances, of



(a) (b) (c)

Fig. 5: **Performance of `RNAinverse` with different seed strategies in violinplots.** For each successfully solved target, 100 $\mathcal{T}$-designs are sampled and the average value is computed for (a) mutation needed from seed to design, (b) normalized ensemble defect of seeds (orange) and designs (blue) to the target structure (the expected distance from target structure to a random structure in the ensemble divided by the structure size), and (c) sequence diversity quantified by the pairwise Hamming distance. In each plot, from left to right, is the seed sequence strategy with separated, biseparated, Boltzmann sampling, Boltzmann sampling with a choice of A and C to fill loop, and uniform sampling.

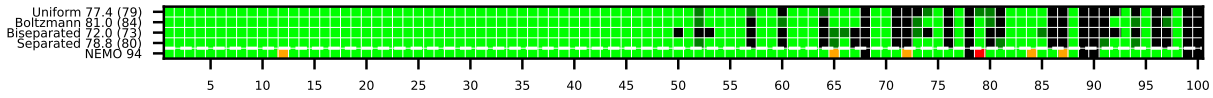


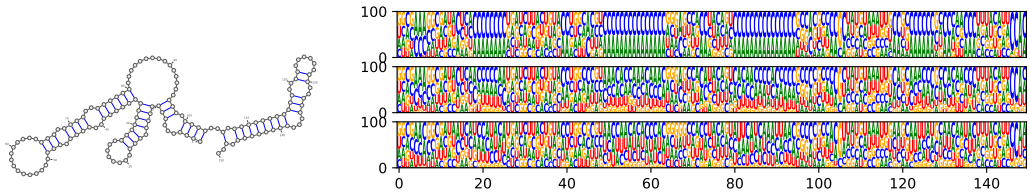
Fig. 6: **Summary of solving Eterna100 v2 benchmark with different `RNAinverse` seed strategies.** Light green indicates success in all 5 runs while dark green means at least one success. From top to bottom are the performance of `RNAinverse` with uniform, Boltzmann, biseparated, and separated seeds. The number is the average amount of solved puzzles for five runs and the one in the parenthesis is the best run result. `NEMO` performance is taken from [10]. Puzzles that are not the only MFE structure of the provided solution are marked orange. Note that the provided `NEMO` solution for puzzle 79 (red) has a different target structure.

which 471 contain at least one isolated base pair. For each seed sequence strategy, we asked for 100 $\mathcal{T}$-designs using `RNAinverse -R-100` with a time limit of 2h, however, most of the tasks were finished within a few minutes (Fig. C.2). In terms of the number of successes, using Boltzmann sampled seeds accomplished all tasks, while 5 and 6 tasks were unfinished with separated and biseparated seeds, showing a limit on the sequence space restriction strategy. However, 100 $\mathcal{T}$-designs were returned for all successful tasks.

Separated seeds enable high stability. We further evaluated the designs among solved structures and visualized the performances in Figure 5. Compared with Boltzmann sampled seeds, separated $\mathcal{B}$-designs are usually closer to the final $\mathcal{T}$-solutions being only one or two mutations away. The resulting $\mathcal{T}$-designs reach low normalized ensemble defect with 0.021 on average, a negative design metric that is often used to quantify the quality of the design. At the same time, using separated seeds results in the lowest sequence diversity among designs.

Biseparated seeds enable high diversity. On the other hand, designing with biseparated seeds has the highest sequence diversity, measured with the average pairwise Hamming distance of obtained $\mathcal{T}$-designs. The sequence diversity is similar to the uniform sampled ones when restricted to paired positions only (Fig. C.4c). The amount of required mutations to find the $\mathcal{T}$-design is close to the Boltzmann sampled seeds with a larger variance while the resulting designs have a higher normalized ensemble defect (0.077 vs 0.040 on average). However, both reach the same level of performance if we consider the best one among 100 $\mathcal{T}$-designs for each structure (Fig. C.4d). To validate the need of `LinearBPDesign` for mixing A and C, we consider alternative Boltzmann sampled seeds where each loop can be filled either with A or C with 50% chance each. The resulting designs have a worse performance regarding these three metrics.

On larger structures, separated seeds only need few mutations. Running the same experiment on larger MFE structures of random sequences of 500 nts yields a similar performance as on the MFE



(a) (b)

Fig. 7: **Example of post-design neutral network traveling.** (a) Structure of interest. (b) Sequence logo of 100 $\mathcal{T}$-designs. From top to bottom, `RNAinverse` solutions with biseparated seeds; resulting sequences after up to 100 moves within neutral network; moving 100 steps within neutral network and ensuring in each step the normalized ensemble defect is at most 0.01 more than the current best with a much longer computational time. The positional dinucleotide entropy for unpaired positions is 1.33, 3.51, 3.58. The average normalized ensemble defect is 0.08, 0.18, 0.08.

structures of 150 nts. `RNAinverse` returned $\mathcal{T}$-designs reaching an average normalized ensemble defect at 0.013 using separated seeds. This value is 0.062 and 0.035 for using, respectively, biseparated and Boltzmann sampled seeds. Unsurprisingly, using both biseparated and Boltzmann sampled seeds requires more mutations for `RNAinverse` to find $\mathcal{T}$-designs (on average 12.1 and 11.2). However, separated seeds for MFE structure of 500 nts still locate close to the final designs with only 2.3 mutations needed on average.

3.3 A naive sampling strategy is sufficient to solve most `EterRNA` 100 puzzles

`EterRNA` 100 v2 benchmark [10] contains 100 artificial puzzles, several of which were intentionally built to be hard to design, by containing a large multiloop or a chain of isolated base pairs and stacks. The benchmark has been often used to validate the performance of RNA design methods based on the number of solved puzzles. To the best of our knowledge, the current leaders are `NEMO` [14] with 94 puzzles [10] using Nested Monte Carlo Search, `eM2dRNAs` with 82 puzzles using a sophisticated structure decomposition [17], and `libLEARNA` with 78 puzzles using reinforcement learning [18]. Note that the negative RNA design problem considered in these methods does not require the target to be the only MFE structure of the solutions. The first two were conducted with a 24h time limit while the latter is unknown.

Success of naive sampling with forced As. We ran `RNAinverse` with each seed strategy 5 times on the benchmark on an Intel Xeon Gold 6342 processor. `RNAinverse` was asked to find one $\mathcal{T}$-design for each puzzle in each run with a two-hour time limit (see Fig. 6 for results). Surprisingly, simply forcing A in unpaired bases, `RNAinverse` is able to solve an average of 77 puzzles using uniform seeds (79 puzzles for best run), increased to 84 with Botzmann-sampled seeds, and to 80 with separated seed for the best run.

Multiloops may be easy to design in $\mathcal{T}$. High degree multiloops are $\mathcal{B}$-undesignable due to the existence of energy-neutral local rearrangements. This "hardness" may not directly imply in $\mathcal{T}$ model as explained in Figure 3. For instance, Puzzle 51 contains a large multiloop of degree 25 (Fig. E.7a). This puzzle can be easily solved with separated seeds with none or few mutations using `RNAinverse`, suggesting that long-range rearrangement has more impact on the "hardness" than multiloop for the Inverse Folding problem in Turner energy model.

Separability over biseparability to get quickly a single solution. `RNAinverse` with biseparated seed can only solve on average 72 puzzles within two hours. An example to illustrate the possible limitation is Puzzle 75, which contains two loops separated by an isolated base pair (Fig. E.7b). Starting with a separated seed sequence where unpaired positions are filling with A, `RNAinverse` managed to find a $\mathcal{T}$-design by mutating two positions next to the isolated base pair. This shows the capacity of `RNAinverse` optimization in response to dangling energy. On the other hand, `RNAinverse` is trapped in the local minimum after six mutations when starting with a separated seed filled with C in unpaired positions.

3.4 Methods on "hard" benchmarks still struggle on "simple" tasks

We also ran `libLEARNA` on the same set consisting of 476 MFE structures used in Section 3.2. For each structure, we gave `libLEARNA` a three-hour time limit and let it run until 100 $\mathcal{T}$-designs were found. `libLEARNA` only successfully returned 100 $\mathcal{T}$-designs for 350 structures. Note that the returned sequences are discarded if a competing MFE structure other than the target exists. Surprisingly, the discarded amount could be 100x more than the number of $\mathcal{T}$-designs for some structures (Fig. C.3). Among the 378 structures with at least 50 $\mathcal{T}$-designs returned, the performance of `libLEARNA` is similar to `RNAinverse` using biseparated seeds with a larger variance in the sequence diversity (Fig. C.4).

4 Discussion

Introducing biseparability combined with `RNAinverse` increases the sequence diversity and the proportion of wobble base pairs GU (1‰ v.s. 3‰), however this may still be far from rational RNA design. Nevertheless, the results of this study demonstrate that negative seeds often are located exactly on or close to the neutral network, allowing fast access to it. Studies on theoretical evolutionary showed that the neutral network of an RNA secondary structure is often quite extensive and sufficiently connected such that simple point mutations can result in long paths enclosed in the network [20]. A similar idea, exploiting this fact to further improve the ensemble defect of designed sequences, is implemented in [27]. As a proof of concept, we chose one of the generated MFE structures as the example and performed a post-design random walk within the neutral network to increase unpaired positions diversity (Fig. 7).

Despite `RNAinverse` being considered outdated and inefficient by now, more than 30 years after its first appearance, we have shown that it is still possible to reach performance comparable to more recent strategies by combining it with negative seeds offered by `LinearBPDesign`. However, the intrinsic strategy of `RNAinverse` still hinders a design of certain target structures. Strategies to improve this could include adapting a more efficient structure decomposition [17] or optimization method. To show this, we tested a naive combination of `LinearBPDesign` (bi)separated seeds with `NEMO` on the popular `EterRNA` benchmark, demonstrating that 88/90 puzzles could be solved within two hours (Fig. E.6). To prevent `NEMO` from exceedingly departing from the initial seed, we implemented a forced restart of `NEMO` every five minutes. Unfortunately, even with this precaution, the distance between the seed and the proposed solution was regularly large enough, to make drawing a decisive conclusion on the impact of negative seed impossible.

The large variance of the performance observed when using biseparated seed hints at a lack of consistency when using uniform sampling. One possible strategy to mitigate this is to add a post-sampling rejection step to filter out unwanted or less probable seeds. However, when using Boltzmann sampled seeds, we detected a significantly lower variance in performance. This suggests that using weighted sampling of `LinearBPDesign`, which supports both positive and negative design paradigms, can further improve the performance. It should be noted that when the base pair energy model is used as done here, this is equivalent to increasing the amount of GC base pairs since the wobble base pair GU is not considered. Using Boltzmann sampling with a stack energy model as was done in [15] requires a more complex dynamic programming scheme, such as provided by the `Infrared` framework [24], to take care of energy contribution from adjacent base pairs as well as the constraints imposed by `LinearBPDesign`.

The success of directly using separated $\mathcal{B}$-designs to solve Inverse Folding within the Turner energy model, notably for the uniformly sampled structure of minimum helix length 3^+, indicates that the structure itself likely exerts a greater influence on the "hardness" of the design than the energy model. Another common "hardness" observed for both energy models is the long-range rearrangement to form an alternative structure. Separability ensures that any alternatives are less favorable in the BP model. The resulting $\mathcal{B}$-designs are found to be close to $\mathcal{T}$-designs with high stability.

Data Availability. The code and scripts to run presented experiments can be found at https://github.com/ViennaRNA/negseeddesign. The different seed initial strategy will be included in the next ViennaRNA release.

Acknowledgments. The authors gratefully acknowledge discussions with Sebastian Will and Laurent Bulteau and specially thank Vladimir Reinharz for offering computational resources for some experiments. HTY is funded by Austrian Science Fund (FWF), grant no. I 4520. LS is funded by FWF no. I 6440-N.

References

1. Andronescu, M., Fejes, A.P., Hutter, F., Hoos, H.H., Condon, A.: A new algorithm for rna secondary structure design. Journal of molecular biology **336**(3), 607–624 (2004)
2. Bonnet, E., Rzazewski, P., Sikora, F.: Designing rna secondary structures is hard. Journal of Computational Biology **27**(3), 302–316 (2020). https://doi.org/10.1089/cmb.2019.0420, https://doi.org/10.1089/cmb.2019.0420, pMID:32160034
3. Boury, T., Bulteau, L., Ponty, Y.: RNA Inverse Folding Can Be Solved in Linear Time for Structures Without Isolated Stacks or Base Pairs. In: Pissis, S.P., Sung, W.K. (eds.) 24th International Workshop on Algorithms in Bioinformatics (WABI 2024). Leibniz International Proceedings in Informatics (LIPIcs), vol. 312, pp. 19:1–19:23. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2024). https://doi.org/10.4230/LIPIcs.WABI.2024.19, https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.WABI.2024.19
4. Cruz, J.A., Westhof, E.: The dynamic landscapes of rna architecture. Cell **136**(4), 604–609 (2009)
5. Delebecque, C.J., Silver, P.A., Lindner, A.B.: Designing and using rna scaffolds to assemble proteins in vivo. Nature protocols **7**(10), 1797–1807 (2012)
6. Hales, J., Héliou, A., Manuch, J., Ponty, Y., Stacho, L.: Combinatorial RNA design: Designability and structure-approximating algorithm in watson-crick and nussinov-jacobson energy models. Algorithmica **79**(3), 835–856 (2017)
7. Hammer, S., Wang, W., Will, S., Ponty, Y.: Fixed-parameter tractable sampling for rna design with multiple target structures. BMC bioinformatics **20**, 1–13 (2019)
8. Hofacker, I.L., Fontana, W., Stadler, P.F., Bonhoeffer, L.S., Tacker, M., Schuster, P., et al.: Fast folding and comparison of rna secondary structures. Monatshefte fur chemie **125**, 167–167 (1994)
9. Isaacs, F.J., Dwyer, D.J., Collins, J.J.: Rna synthetic biology. Nature biotechnology **24**(5), 545–554 (2006)
10. Koodli, R.V., Rudolfs, B., Wayment-Steele, H.K., Designers, E.S., Das, R.: Redesigning the eterna100 for the vienna 2 folding engine. BioRxiv pp. 2021–08 (2021)
11. Luo, D.: From biology to materials: engineering dna and rna for drug delivery and nanomedicine. Advanced drug delivery reviews **6**(62), 591 (2010)
12. Lyngsø, R.B., Anderson, J.W., Sizikova, E., Badugu, A., Hyland, T., Hein, J.: Frnakenstein: multiple target inverse rna folding. BMC bioinformatics **13**, 1–12 (2012)
13. Nussinov, R., Jacobson, A.B.: Fast algorithm for predicting the secondary structure of single-stranded rna. Proceedings of the National Academy of Sciences **77**(11), 6309–6313 (1980)
14. Portela, F.: An unexpectedly effective monte carlo technique for the rna inverse folding problem. BioRxiv p. 345587 (2018)
15. Reinharz, V., Ponty, Y., Waldispühl, J.: A weighted sampling algorithm for the design of rna sequences with targeted secondary structure and nucleotide distribution. Bioinformatics **29**(13), i308–i315 (2013)
16. Retwitzer, M.D., Reinharz, V., Churkin, A., Ponty, Y., Waldispühl, J., Barash, D.: incaRNAfbinv 2.0: a webserver and software with motif control for fragment-based design of RNAs. Bioinformatics **36**(9), 2920–2922 (01 2020). https://doi.org/10.1093/bioinformatics/btaa039, https://doi.org/10.1093/bioinformatics/btaa039
17. Rubio-Largo, Á., Lozano-García, N., Granado-Criado, J.M., Vega-Rodríguez, M.A.: Solving the rna inverse folding problem through target structure decomposition and multiobjective evolutionary computation. Applied Soft Computing p. 110779 (2023)
18. Runge, F., Franke, J., Fertmann, D., Backofen, R., Hutter, F.: Partial rna design. Bioinformatics **40**(Supplement_1), i437–i445 (2024)
19. Santosh, B., Varshney, A., Yadava, P.K.: Non-coding rnas: biological functions and applications. Cell biochemistry and function **33**(1), 14–22 (2015)
20. Schuster, P., Fontana, W., Stadler, P.F., Hofacker, I.L.: From sequences to shapes and back: a case study in rna secondary structures. Proceedings of the Royal Society of London. Series B: Biological Sciences **255**(1344), 279–284 (1994)
21. Shi, J., Das, R., Pande, V.S.: Sentrna: Improving computational rna design by incorporating a prior of human design strategies. arXiv preprint arXiv:1803.03146 (2018)
22. Turner, D.H., Mathews, D.H.: Nndb: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. Nucleic Acids Research **38**(suppl_1), D280–D282 (10 2009). https://doi.org/10.1093/nar/gkp892
23. Yao, H.T., Chauve, C., Regnier, M., Ponty, Y.: Exponentially few rna structures are designable. In: Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics. p. 289–298. BCB '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3307339.3342163, https://doi.org/10.1145/3307339.3342163
24. Yao, H.T., Marchand, B., Berkemer, S.J., Ponty, Y., Will, S.: Infrared: a declarative tree decomposition-powered framework for bioinformatics. Algorithms for Molecular Biology **19**(1), 13 (2024)

25. Yao, H.T., Waldispühl, J., Ponty, Y., Will, S.: 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)
26. Zhang, H., Zhang, L., Lin, A., Xu, C., Li, Z., Liu, K., Liu, B., Ma, X., Zhao, F., Jiang, H., Chen, C., Shen, H., Li, H., Mathews, D.H., Zhang, Y., Huang, L.: Algorithm for optimized mrna design improves stability and immunogenicity. Nature **621**(7978), 396–403 (May 2023). https://doi.org/10.1038/s41586-023-06127-z
27. Zhou, T., Dai, N., Li, S., Ward, M., Mathews, D.H., Huang, L.: Rna design via structure-aware multifrontier ensemble optimization. Bioinformatics **39**(Supplement_1), i563–i571 (2023)
28. Zhou, T., Tang, W.Y., Mathews, D.H., Huang, L.: Undesignable rna structure identification via rival structure generation and structure decomposition. In: Ma, J. (ed.) Research in Computational Molecular Biology. pp. 270–287. Springer Nature Switzerland, Cham (2024)