Combinatorial RNA Design: Designability and Structure-Approximating Algorithm

Jozef Haleš, Ján Maňuch, Yann Ponty, Ladislav Stacho

Combinatorial RNA Design: Designability and Structure-Approximating Algorithm

Jozef Haleš[1], Ján Maňuch[1,3], Yann Ponty[1,2], and Ladislav Stacho[1]

[1] Department of Mathematics, Simon Fraser University, Canada
[2] Pacific Institute for Mathematical Sciences, CNRS UMI3069, Canada
[3] Department of Computer Science, University of British Columbia, Canada

Abstract. In this work, we consider the *Combinatorial RNA Design problem*, a *minimal* instance of the RNA design problem which aims at finding a sequence that admits a given target as its unique base pair maximizing structure. We provide complete characterizations for the structures that can be designed using restricted alphabets.
Under a classic four-letter alphabet, we provide a complete characterization of designable structures without unpaired bases. When unpaired bases are allowed, we provide partial characterizations for classes of designable/undesignable structures, and show that the class of designable structures is closed under the stutter operation. Membership of a given structure to any of the classes can be tested in linear time and, for positive instances, a solution sequence can also be generated in linear time.
Finally, we consider a structure-approximating version of the problem that allows to extend bands (helices) and, assuming that the input structure avoids two motifs, we provide a linear-time algorithm that produces a designable structure with at most twice more base pairs than the input structure.

1 Introduction

RiboNucleic Acids (RNAs) are biomolecules which act in almost every aspect of cellular life, and can be abstracted as a sequence of nucleotides, i.e., a string over the alphabet $\{\mathsf{A}, \mathsf{U}, \mathsf{C}, \mathsf{G}\}$. Due to their versatility, and the specificity of their interactions, they are increasingly being used as therapeutic agents [21], and as building blocks for the emerging field of synthetic biology [16, 18]. A substantial proportion of the functional roles played by RNA rely on interactions with other molecules to activate/repress dynamical properties of some biological system, and ultimately require the adoption of a specific conformation. Accordingly, RNA bioinformatics has dedicated much effort to developing energy models [13, 20] and algorithms [14, 24] to predict the secondary structure of RNA, a combinatorial description of the conformation adopted by an RNA which only retains interacting positions, or base pairs. Historically, structure prediction has been addressed as an optimization problem, whose expected output is a secondary structure which minimizes some notion of free-energy [14, 24]. The performances of the RNA folding prediction problem have now reached a point where *in silico* predictions have become generally reliable [13], allowing for large scale studies and fueling the discovery of an increasing number of functional families [8].

Due to the existence of expressive, yet tractable, energy models, coupled with promising applications in multiple fields (pharmaceutical research, natural computing, biochemistry...), a wide array of computational methods [9, 3, 1, 4, 2, 19, 22, 11, 12, 7, 10, 15, 23, 5] have been proposed to tackle the natural inverse version of the structure prediction, the RNA design problem. In this problem, one attempts to perform the *in silico* synthesis of artificial RNA sequences, performing a predefined biological function *in vitro* or *in vivo*. Given the prevalence of structure in the function of an RNA, one of the foremost goal of RNA design (sometimes named inverse folding in the literature) is that the designed sequence should fold into a predefined secondary structure. In other words, it should not be challenged by alternative stable structures having similar or lower free-energy.

Despite a rich, fast-growing, body of literature dedicated to the problem, there is currently no exact polynomial-time algorithm for the problem. Moreover, the complexity of the problem remains

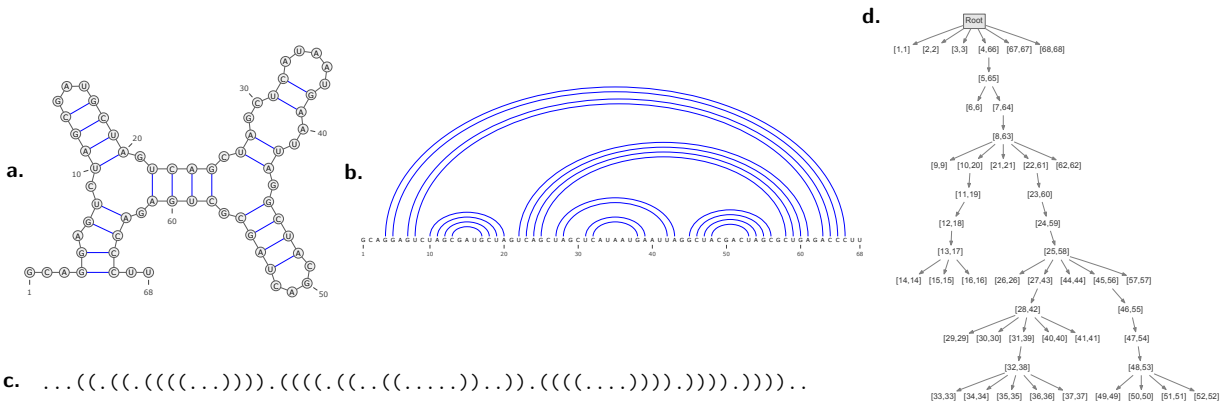

Fig. 1. Four equivalent representations for an RNA secondary structure of length 68, consisting of 20 base pairs forming 7 bands: outer-planar graph (a.), arc-annotated representation (b.), parenthesized expression (c.), and tree representation (d.)

open (see Section 5 for details). It can be argued that this situation, quite exceptional in the field of computational biology, partly stems from the intricacies of the Turner free-energy model [20] which associates experimentally-determined energy contributions to $\sim 2.4\times10^4$ structure/sequence motifs. This motivates a reductionist approach, where one studies an idealized version of the RNA design problem, lending itself to algorithmic intuitions, while hopefully retaining the presumed difficulty of the original problem.

In this work, we introduce the *Combinatorial RNA Design problem*, a *minimal* instance of the RNA design problem which aims at finding a sequence that admits the target structure as its unique base pair maximizing structure. After this short introduction, Sec. 2 states definitions and problems. In Sec. 3, we summarize our results, some of which are proven in Sec. 4. Finally, we conclude in Sec. 5 with some remarks, open problems and future extensions of this work.

2 Definitions and notations

RNA secondary structure. An RNA can be encoded as a sequence of nucleotides, i.e., a string $w = w_1 \cdots w_{|W|} \in \{\mathsf{A}, \mathsf{U}, \mathsf{C}, \mathsf{G}\}^\star$. The prefix of w of length i is denoted as $w_{[1,i]}$ and $|w|_b$ denotes the number of occurrences of b in w. A (pseudoknot-free) secondary structure S on an RNA of length n is a pair (n, P), where P is a set of base pairs $\{(l_i, r_i)\}_{i=1}^{p} \subset [1, n]^2$ such that:

- $\forall i \in [1, p]$, $l_i < r_i$;
- $\forall i, j \in [1, p], l_i \neq l_j$, $l_i \neq r_j$, $r_i \neq r_j$ (each position is involved in at most one base pair);
- $\nexists i, j \in [1, p]$, $l_i < l_j < r_i < r_j$ (base pairs (l_i, r_i) and (l_j, r_j) do not cross).

The set of all secondary structures is denoted by $\mathcal{S}$, and its restriction to structures of length n by $\mathcal{S}_n$. The unpaired positions U_S in a secondary structure $S = (n, P)$ is the set of indices $k \in [1, n]$ that are not involved in a base pair. A structure S is saturated if $U_S = \emptyset$. Given a sequence w and a structure $S = (|w|, P)$, let $u_i = \varepsilon$ if $i \in U_S$ and $u_i = w_i$, otherwise, where ε is the empty sequence. Define the S-paired restriction of w (paired restriction of S), denoted as $\mathsf{Paired}(w, S)$ ($\mathsf{Paired}(S)$), as $u_1 \cdots u_{|w|}$ (respectively, $\{(|u_1 \cdots u_i|, |u_1 \cdots u_j|) \mid (i, j) \in P\}$). A maximal subset $B = \{(i, j), (i + 1, j - 1), \ldots, (i + \ell, j - \ell)\}$ of P for some integer i, j, ℓ is called a band (sometimes referred to as helix or stem) of size $\ell = |B|$, of $S = (n, P)$. Note that every base pair belongs to exactly one band.

Dot-parentheses notation. A well-parenthesized sequence $s \in \{(,),.\}^*$ can be used to represent a secondary structure. There is one-to-one correspondence between secondary structures and such well-parenthesized sequences: any base pair $(l, r) \in S$ becomes a pair of corresponding opening and closing parentheses in s at position l and r respectively ($s_l = ($ and $s_r =)$), and any unpaired position i corresponds to a dot ($s_i = .$).

k-stutter. The k-stutter of a sequence s, denoted by $s^{[k]}$ is the result of an independent copy k-times of each of the characters in s. This operation can be applied to both RNA sequences and structures in the dot-parentheses notation.

Tree representation. Alternatively, the tree representation, denoted by T_S, for $S = (n, P)$ is a rooted ordered tree whose vertex set V_S consists of intervals $[l, r]$ for any base pair $(l, r) \in P$, and $[k, k]$ for every $k \in U_S$. A virtual root $[0, n + 1]$ is added for convenience. Each $[k, k]$ node is called unpaired node, all other nodes (including the root) are called paired nodes. The children of an interval $I \in V_S$ are the maximal proper subintervals $I' \in V_S$ of I ordered by the left points of the intervals. The degree of a vertex $I \in V_S$ is the total number of its paired neighbors, including its parent (if any). We denote by $D(S)$ the maximal degree of nodes in T_S.

Proper, greedy and separated coloring of the tree representation. Consider the tree representation T_S of structure S. Color every paired node of T_S different from the root by black, white or grey color. This coloring is called proper if:

1. every node has at most one black, at most one white and at most two grey children;
2. a node with color c has at most one child with color c;
3. a black node does not have a white child and a white node does not have a black child.

A greedy coloring of T_S is the coloring obtained by recursive application of the following rule starting from the root and continuing towards leaves: if the node is black, color the first paired child black and the remaining paired children grey, if the node is white, color the first paired child white and the remaining paired children grey, otherwise (the grey node or the root), color the first paired child black, second white and the remaining paired children grey. It is easy to check that if the degree of each node is at most four then the greedy coloring is a proper coloring.

Given a proper coloring of T_S, let the level of each node be the number of black nodes minus the number of white nodes on the path from this node to the root. A proper coloring is called separated if the two sets of levels, associated with grey and unpaired nodes respectively, do not overlap.

2.1 Statement of the generic RNA design problem

Consider an energy model $\mathcal{M}$, which associates a free-energy $E_{\mathcal{M}}(w, S) \in \mathbb{R}^- \cup \{+\infty\}$ to each secondary structure $S \in \mathcal{S}_{|w|}$ for a given RNA sequence w. The minimum free-energy (MFE) structure prediction problem is typically defined as follows:

RNA-FOLD$_{\mathcal{M}}$ problem
Input: RNA sequence w
Output: $S^\star_{\mathcal{M}}(w) := \operatorname{argmin}_{S' \in \mathcal{S}_{|w|}} E_{\mathcal{M}}(w, S')$.

The existence of competing structures, having comparable or lower free-energy for a given RNA, impacts the well-definedness of the folding process. The detection of such situations is therefore of interest, and can be rephrased as follows:

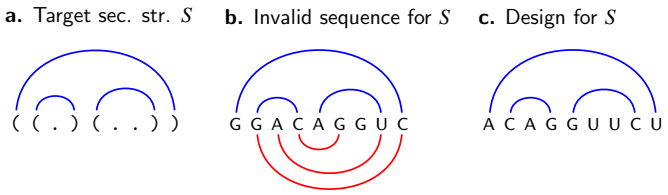

Fig. 2. The combinatorial RNA design problem: Starting from a secondary structure S (a.), our goal is to design an RNA sequence which uniquely folds, with maximum number of base pairs, into S. The sequence proposed in b. is invalid due to the existence of an alternative structure (lower half-plane, red) having the same number of base pairs as S. The right-most sequence (c.) is a design for S.

UNIQUE-FOLD$_{\mathcal{M}}$ problem
Input: Sequence w + Energy distance $\Delta > 0$
Output: True if, for every $S' \in \mathcal{S}_{|w|} \setminus \{S^{\star}_{\mathcal{M}}(w)\}$, $E_{\mathcal{M}}(w, S') \geq E_{\mathcal{M}}(w, S^{\star}_{\mathcal{M}}(w)) + \Delta$.
False otherwise.

We can now define the combinatorial RNA Design problem as:

RNA-DESIGN$_{\mathcal{M},\Sigma}$ problem
Input: Secondary structure S + Energy distance $\Delta > 0$
Output: RNA sequence $w \in \Sigma^{\star}$ — called an $(\mathcal{M}, \Sigma, \Delta)$-design for S — such that:

$$\text{RNA-FOLD}_{\mathcal{M}}(w) = S \quad \text{and} \quad \text{UNIQUE-FOLD}_{\mathcal{M}}(w, \Delta),$$

or $\varnothing$ if no such sequence exists.

Structures for which there exists an $(\mathcal{M}, \Sigma, \Delta)$-design are called $(\mathcal{M}, \Sigma, \Delta)$-designable. Let Designable$(\mathcal{M}, \Sigma, \Delta)$ be the set of all such structures. If it is clear from the context, we will usually drop $\mathcal{M}$, Σ and/or Δ.

2.2 Combinatorial design in a simple base pair energy model

In this work, we adopt a Watson-Crick energy model $\mathcal{W}$, which only allows pairs involving complementary letters, i.e., in $\{\mathsf{C}, \mathsf{G}\}$ and $\{\mathsf{A}, \mathsf{U}\}$.

Definition 1 (Watson-Crick energy model $\mathcal{W}$).

$$E_{\mathcal{W}}(w, S) = \begin{cases} -|S| & \text{if } \forall (l, r) \in S,\ w_l \text{ is complementary with } w_r, \\ +\infty & \text{otherwise.} \end{cases}$$

We say that the structure is compatible with a sequence w, if $E_{\mathcal{W}}(w, S) < +\infty$.

Minimizing $E_{\mathcal{W}}(w, S)$ is equivalent to maximizing $|S|$, thus RNA-FOLD$_{\mathcal{W}}$ is a classic base pair maximization problem. It can be solved by dynamic programming, historically in $\mathcal{O}(n^3)$ complexity [14], or in $\mathcal{O}(n^3/\log(n))$ current best time complexity [6]. A backtracking procedure reconstructs the MFE structure, and can be easily adapted to assess the uniqueness of the MFE structure.

3 Statement of the results

We consider the design problem in a base pairing energy model $\mathcal{W}$ restricted to Watson-Crick base pairs $\{\mathsf{C}, \mathsf{G}\}$ and $\{\mathsf{A}, \mathsf{U}\}$. We set $\Delta = 1$, which forbids designed sequence to adopt alternative structures having greater or equal number of base pairs than the target structure. Let us first characterize the sets Designable(Σ) of designable structures over partial alphabets Σ. Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

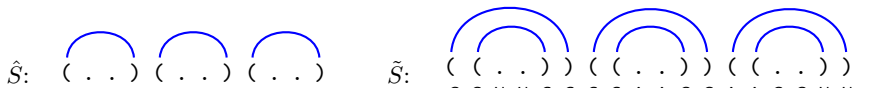

Fig. 3. An example of undesignable (left) and designable structure (right).

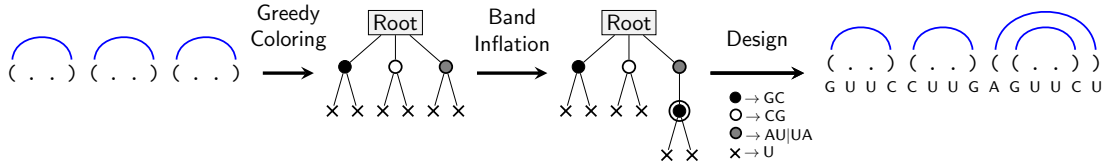

Fig. 4. Application of the structure-approximating algorithm to the non-designable structure $\hat{S}$ in Fig. 3: A base pair (circled black node) is inserted in the greedily colored tree, offsetting the levels of white and unpaired nodes (crosses) to even and odd levels respectively, so that the resulting tree is proper/separated, representing a designable structure.

Designability over restricted alphabets.

R1 For every $u \in \mathbb{N}^+$, $\mathsf{Designable}(\Sigma_{0,u}) = \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;
R2 $\mathsf{Designable}(\Sigma_{1,0}) = \{S \in \mathcal{S} \mid S \text{ is saturated and } D(S) \leq 2\} \cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;
R3 $\mathsf{Designable}(\Sigma_{1,1}) = \{S \in \mathcal{S} \mid D(S) \leq 2\}$.

Designability over the complete alphabet $\Sigma_{2,0} = \{\mathsf{A}, \mathsf{U}, \mathsf{C}, \mathsf{G}\}$.

R4 $\mathsf{Designable}(\Sigma_{2,0}) \cap \{S \in \mathcal{S} \mid S \text{ is saturated}\} = \{S \in \mathcal{S} \mid D(S) \leq 4\} \cap \{S \in \mathcal{S} \mid S \text{ is saturated}\}$.

When unpaired positions are allowed in the target structure, our characterization is only partial:

R5 Let m_5 represent "a node having degree more than four", and $m_{3\circ}$ be "a node having one or more unpaired children, and degree greater than two", then

$$\mathsf{Designable}(\Sigma_{2,0}) \cap \{S \in \mathcal{S} \mid S \text{ contains } m_5 \text{ or } m_{3\circ}\} = \emptyset\,;$$

R6 Let Sep be the set of structures for which there exists a separated (proper) coloring of the tree representation, then $\mathsf{Sep} \subset \mathsf{Designable}(\Sigma_{2,0})$;
R7 The set of $\Sigma_{2,0}$-designable structures is closed under the k-stutter operations:

$$\forall S \in \mathcal{S}, \forall k \in \mathbb{N}^+ : \quad S \in \mathsf{Designable}(\Sigma_{2,0}) \implies S^{[k]} \in \mathsf{Designable}(\Sigma_{2,0})\,.$$

We note that $S^{[k]} \in \mathsf{Designable}(\Sigma_{2,0})$ does not imply that $S \in \mathsf{Designable}(\Sigma_{2,0})$. For instance, it can be verified that $\hat{S}^{[2]}$ is $\Sigma_{2,0}$-designable, while $\hat{S}$ is not, cf. Figure 3. Membership to the classes described in **R1-R5** can be tested by trivial linear-time algorithms, which can also be adapted into linear-time algorithms for the $\mathsf{RNA\text{-}DESIGN}_{\mathcal{M},\Sigma}$ problem.

Structure-approximating algorithm. Unfortunately, the absence of m_5 or $m_{3\circ}$, while necessary, is generally not sufficient to ensure designability. For instance, $\hat{S}$ in Figure 3 clearly does not contain m_5 or $m_{3\circ}$, yet cannot be designed. In such cases, the unwanted interactions can be penalized by the duplication of some base pairs. For instance, duplicating the base pairs in the above example yields $\Sigma_{2,0}$-designable structure $\tilde{S}$.

R8 Any structure S without m_5 and $m_{3\circ}$ can be transformed in $\Theta(n)$ time into a $\Sigma_{2,0}$-designable structure S', by inflating a subset of its base pairs (at most one per band) so that the greedy coloring of the resulting structure is proper and separated, as illustrated by Figure 4.

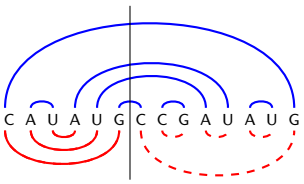

Fig. 5. Construction the saturated structure compatible with the suffix v. The vertical line splits the sequence into a prefix u and a suffix v. Blue and red arcs depict saturated structures compatible with w and u respectively. Dashed red arcs represent the induced saturated structure compatible with v.

4 Proofs

R1 is trivial since, in the absence of complementary letters, the structures without base pairs are the only structures whose energy is not infinite.

Theorem 1 (Result R4). *A saturated sec. str. S is $\Sigma_{c,0}$-designable if and only if $D(S) \leq 2c$.*

Proof. First, we will show that the degree condition is necessary. Assume to the contrary that $D(S) > 2c$ and S has a design w. Let $[a, b]$ be a vertex with degree $d \geq 2c+1$ in T_S. Let $\{[l_i, r_i]\}_{i=1}^{d}$ be the (paired) children of $[a, b]$ and the node $[a, b]$ if $[a, b]$ is not the root. Let $L_i = l_i$ and $R_i = r_i$ if $[l_i, r_i]$ is a child of $[a, b]$, and $L_i = r_i$ and $R_i = l_i$ if it is $[a, b]$. Then among bases $w_{L_1}, \ldots, w_{L_d}$ must be a pair of repeated letters. Let $w_{L_i} = w_{L_j}$ be such a pair with $L_i < L_j$. It is easy to check that $S \setminus \{(l_i, r_i), (l_j, r_j)\} \cup \{(L_i, R_j), (R_i, L_j)\}$ is a structure compatible with w with the same number of base pairs as S, a contradiction with the assumption that w is a design for S.

To show that the degree condition is also sufficient, we need further definitions and claims. First, we say that a sequence $w \in \Sigma^*$ is saturable if there is a saturated structure compatible with w. Note that the concatenation of two saturable sequences is also saturable. Then the following claim characterizes the cases when a saturable sequence can be split into saturable sequences.

Claim 1.1. *Let $w = uv$ be a saturable sequence of length k. If u is saturable, then so is v.*

Proof. Consider a saturated structure S compatible with sequence w and saturated structure S_u compatible with u. We will construct a saturated structure S_v compatible with v.

Consider a graph G with vertex set $\{1, \ldots, k\}$ and edge set defined by pairs in $S \cup S_u$. Obviously, this graph is a collection of alternating paths (alternating between pairs from S and from S_u, starting and ending with positions in v) and alternating cyclic paths, and it has a planar embedding such that all vertices lie on a line in their order: pairs in S are drawn as non-crossing arcs above the line and pairs in S_u as non-crossing arcs below the line. Note that every position in v is an end-point of exactly one path in the collection.

Define set of base pairs S_v by pairing the end-points of the paths in G, cf. Figure 5. We will show that S_v is a structure. Consider a graph G' constructed by adding pairs in S_v to G. This graph is a collection of cyclic paths. Consider an embedding of G' into plane that extends the planar embedding of G by adding arcs corresponding to the pairs in S_v below the line containing all the vertices. If two base pairs $b, b' \in S_v$ cross then the cyclic path containing b and the cyclic path containing b' intersect in exactly one point. By Jordan's curve theorem, this is a contradiction. It follows that S_v is a saturated structure, and hence v is also saturable. □

We define w to be an atomic saturable sequence if no proper prefix of w is saturable. Clearly, every saturated structure compatible with an atomic saturable sequence w contains the base pair $(1, |w|)$. On the other hand, by Claim 1.1, if every saturated structure compatible with w contains the pair $(1, |w|)$, then w is an atomic saturable sequence. A design w that is also an atomic saturable sequence will be called an atomic saturable design. A concatenation of two or more atomic saturable designs is obviously not an atomic saturable sequence and it is not necessarily a design. However, we have the following claims.

Claim 1.2. *The concatenation of t atomic saturable designs $w^1 \dots w^t$ for structures $S^1 \dots S^{|t|}$, such that $w^i_1 \neq w^j_1, \forall 1 \leq i < j \leq t$, is a design for the concatenated (saturated) structure $S = S^1 \cdots S^{|t|}$.*

Proof. Assume that $W := w^1 \cdots w^t$ is not a design, then there exist a saturated structure $S' \neq S$ for W. We show that positing such an alternative structure leads to a contradiction, reminding that each S^i is saturated and contains a pair $(1, |w^i|)$. Assume that there exists a leftmost word w_i, $i \in [1, |t|]$, such that w^i_1 is not paired with $w^i_{|w^i|}$ in S'. If w^i_1 is not paired, then S' is not saturated, a contradiction. Let w^j_k, $j \geq i$, be the partner of w^i_1 in S', and let $u := w^i \cdots w^{j-1} w^j_{[1,k]}$. If $k = |w^j|$, then $j > i$ and, by complementarity, $w^i_1 = w^j_1$ which contradicts the preconditions. Hence, we can assume that $k < |w^j|$. Since u and each of the $w^i, \dots, w^{j-1}$ are saturable, by iterated application of Claim 1.1, we conclude that $v = w^j_{[1,k]}$ is saturable as well and, from Claim 1.1, so is $v' = w^j_{[k+1,|w^j|]}$. This contradicts the hypothesis that $w^j = v.v'$ is an atomic saturable design, since there exists a saturated folding for w^j which does not pair its extremities. Consequently, S' pairs the first and last letters in each w^k, hence $S' = S$ since each w^i is a design, again a contradiction. We conclude that no alternative saturated folding exists for W, i.e. W is a design for S. □

Claim 1.3. *Consider t atomic saturable designs $w^1 = w^1_1 \cdots w^1_{|w^1|}$, …, $w^t = w^t_1 \cdots w^t_{|w^t|}$ and a pair a, b of complementary letters such that $w^i_1 \neq b$ for every $1 \leq i \leq t$ and $w^i_1 \neq w^j_1$ for every $1 \leq i < j \leq t$. Then $W = a w^1 \cdots w^t b$ is an atomic saturable design.*

Proof. We will first show that W is an atomic saturable sequence. Assume to the contrary that there is a proper prefix of W that is saturable. Consider the shortest such prefix $a w^1 \cdots w^i w^{i+1}_{[1,j]}$. Obviously, a has to be paired with w^{i+1}_j, otherwise we can find a shorter saturable prefix. This implies that $b = w^{i+1}_j$ and that $w^1 \cdots w^i w^{i+1}_{[1,j-1]}$ is saturable as well. By repeated application of Claim 1.1, we have that $w^{i+1}_{[1,j-1]}$ is saturable. Since it is a prefix of atomic saturable sequence w^{i+1}, it must be the empty sequence, i.e., $j = 1$. Therefore, $b = w^{i+1}_1$, a contradiction with the assumptions of the claim. Thus, W is an atomic saturable sequence.

Now we will show that W is a design. Consider any MFE (saturated) structure S for W. Since W is atomic saturable, a is paired with b in S. By Claim 1.2, $w^1 \cdots w^t$ is a design. It follows that W is a design as well. □

To prove the sufficiency of the degree condition, consider the following algorithm, which takes as input a saturated structure S with $D(S) \leq 2c$, and returns a design w for S:

- Let $\{[l_i, r_i]\}_{i=1}^d$ be the children of the root. Assign to each w_{l_i}, w_{r_i} complementary bases such that $\forall 1 \leq i < j \leq d: \ w_{l_i} \neq w_{l_j}$.
- While there exists an unprocessed internal node $[a, b]$ whose parent has been processed (if there is no such node, stop and return w). Let $\{[l_i, r_i]\}_{i=1}^d$ be the children of $[a, b]$. Assign to each w_{l_i}, w_{r_i} complementary bases such that $\forall 1 \leq i \leq d: \ w_{l_i} \neq w_a$ and $\forall 1 \leq i < j \leq d: \ w_{l_i} \neq w_{l_j}$.

Note that since the alphabet contains c pairs of complementary bases, the assignment at each step of the algorithm is possible. We will show that the returned sequence w is a design for S. We will show by tree induction on the size subtrees that $w_i \cdots w_j$ is an atomic saturable design for every internal node $[i, j]$. It is easy to check that this is satisfied at the leaves. Consider an internal node u. By the induction hypothesis, sequences for each child subtree of u are atomic saturable designs. Furthermore, by the choice of bases at children nodes of u, all assumptions of Claim 1.3 are satisfied, hence, the sequence for node u is also an atomic saturable design. The claim holds. Finally, we can apply Claim 1.2 at the root, which yields that w is a design. □

Corollary 2 (Result R2). *A structure S is $\Sigma_{1,0}$-designable if and only if it does not contain any base pairs, or it is saturated and $D(S) \leq 2$.*

Proof. If S contains a base pair and an unpaired position, then it can be easily checked that S is not $\Sigma_{1,0}$-designable. Hence, any $\Sigma_{1,0}$-designable structure is either empty, and trivially designable using a single letter, or saturated. In the latter case, by Theorem 1, we know that designable structures are exactly those that are saturated, and such that $D(S) \leq 2$. The claim follows. □

Corollary 3 (Result R3). *A structure S is $\Sigma_{1,1}$-designable if and only if $D(S) \leq 2$.*

Proof. First, suppose S is $\Sigma_{1,1}$-designable and let w be a design for S. Then $\mathsf{Paired}(w, S)$ is a design for $\mathsf{Paired}(S)$. Since the paired restriction $\mathsf{Paired}(S)$ is saturated, it is over alphabet $\Sigma_{1,0} \subset \Sigma_{1,1}$, and by Theorem 1, $D(\mathsf{Paired}(S)) \leq 2$. Hence, $D(S) = D(\mathsf{Paired}(S)) \leq 2$.

Conversely, suppose that $D(S) \leq 2$. Construct a design for S as follows. Since $\mathsf{Paired}(S)$ is saturated, by Theorem 1, there is a design $\bar{w}$ for $\mathsf{Paired}(S)$ over $\Sigma_{1,0} \subset \Sigma_{1,1}$. Construct w from $\bar{w}$ by inserting the base without a complementary base at every unpaired position of S. Let S' be an MFE structure for w. Obviously, all unpaired positions in S are also unpaired in S'. We must have $\mathsf{Paired}(S') = \mathsf{Paired}(S)$, otherwise we have an alternative structure for $\bar{w}$, a contradiction. Hence, $S' = S$, i.e., w is a design for S. □

Result **R4** follows readily from Theorem 1 by taking $c = 2$.

Lemma 4 (Result R5). *Any structure that contains m_5 or $m_{3\circ}$ is not $\Sigma_{2,0}$-designable.*

Proof. Assume that S is $\Sigma_{2,0}$-designable and let w be a design for S. Then $\mathsf{Paired}(w, S)$ is a design for $\mathsf{Paired}(S)$. Since $\mathsf{Paired}(S)$ is saturated, by Theorem 1, $D(S) = D(\mathsf{Paired}(S)) \leq 4$, hence, S cannot contain motif m_5. Now, assume to the contrary that S contain motif $m_{3\circ}$ appearing at node $[a, b]$ of T_S. Let $\{[l_i, r_i]\}_{i=1}^{3}$ be some paired children of $[a, b]$ and the node $[a, b]$ if $[a, b]$ is not the root, and $[u, u]$ an unpaired child of $[a, b]$. Let $L_i = l_i$ and $R_i = r_i$ if $[l_i, r_i]$ is a child of $[a, b]$, and $L_i = r_i$ and $R_i = l_i$ if it is $[a, b]$. If among bases $w_{L_1}, \ldots, w_{L_3}$ there is a pair of repeated letters, then we can construct an alternative MFE structure for w (see the first paragraph in the proof of Theorem 1). Assume that these three bases are different. Then for some $i = 1, 2, 3$, w_u equals either w_{l_i} or w_{r_i}, say it equals w_{l_i}. Then $S \setminus \{(l_i, r_i)\} \cup \{(u, r_i)\}$ is an MFE structure for S, a contradiction with the assumption that w is a design for S. □

Theorem 5 (Result R6). *If the tree representation of a structure S admits a separated coloring then S is $\Sigma_{2,0}$-designable.*

Proof. Given a sequence w, we define the level $L(i)$ of position i as $L(i) = |w_{[1,i]}|_{\mathsf{G}} - |w_{[1,i]}|_{\mathsf{C}}$.

Claim 5.1. *Consider any structure compatible with sequence w that contains some $\mathsf{A}-\mathsf{U}$ base pair between positions at different levels, then some G or C is left unpaired.*

Proof. Consider that the $\mathsf{A}-\mathsf{U}$ base pair occurs at position (a, b), and note that the bases of the substring $w_{[a+1,b-1]}$ can only base pair among themselves without introducing crossings. We will show that G's and C's are not balanced in this substring. Since $w_b \in \{\mathsf{A}, \mathsf{U}\}$, $L(b) = L(b-1)$. Hence, by the definition of L, we have that

$$|w_{[a+1,b-1]}|_{\mathsf{G}} - |w_{[a+1,b-1]}|_{\mathsf{C}} = L(b-1) - L(a) = L(b) - L(a) \neq 0\,.$$

Therefore, at least one G or C in the substring remains unpaired in this structure. □

Consider a separated coloring of the tree representation of S. We will use this coloring to construct a design w for S, by specifying a nucleotide at each position of w. First, for each unpaired position i, set $w_i = \mathsf{U}$. Second, apply a modified version of the algorithm described in Theorem 4 to set the bases of paired positions in which black nodes are assigned to base pair $\mathsf{G}-\mathsf{C}$, white nodes to $\mathsf{C}-\mathsf{G}$ and grey nodes to $\mathsf{A}-\mathsf{U}$ or $\mathsf{U}-\mathsf{A}$. The algorithm ignores unpaired nodes in the tree representation of S. Since the coloring is proper such assignment is always possible at every step of the algorithm. We claim that for any node $[i,j]$ (paired or unpaired), the level of position i is the same as the level of the node $[i,j]$. To verify this, observe that the substring of w corresponding to any subtree has the same number of G's and C's. Hence, for any node $[i,j]$, the level of position i depends only on nodes on the path from this node to the root. It is easy to check that the level of i is equal to the level the node. Note that if $[i,j]$ is a grey node then the level of position j is the same as the level of i, i.e., the same as the level of $[i,j]$.

We will show that the constructed w is a design for S. Since all C's and A's of w are paired in S, S is an MFE structure for w. We need to show that it is the only MFE structure for w. Consider an MFE structure S' for w different from S. Since w has the same number of G's and C's, S' must pair all G's, C's and A's of w. We will show that that all unpaired positions in S are also unpaired in S'. Assume to the contrary that position i is unpaired in S, but it is paired to j in S'. We must have $w_i = \mathsf{U}$ and $w_j = \mathsf{A}$. Since the coloring is separated, the unpaired node $[i,i]$ has a different level than the grey node containing j, and hence, the level of i is different from the level of j. It follows by Claim 5.1 that some G or C is unpaired in S', a contradiction. Consider paired restrictions of S, S' and w. Both $\mathsf{Paired}(S)$ and $\mathsf{Paired}(S')$ are saturated and compatible with $\mathsf{Paired}(w,S)$ and they are different since S and S' are different and agree on the unpaired positions. Furthermore, $\mathsf{Paired}(w,S)$ can be produced by the algorithm described in Theorem 4 for the input structure $\mathsf{Paired}(S)$, and hence, by Theorem 1, $\mathsf{Paired}(w,S)$ is a design for $\mathsf{Paired}(S)$, which contradicts the existence of $\mathsf{Paired}(S')$. Hence, w is a design for S. □

Theorem 6 (Result R7). *If w is a design for a structure S, then for any integer $k \geq 1$, $w^{[k]}$ is a design for $S^{[k]}$. In particular, if a structure S is $\Sigma_{2,0}$-designable, then so is $S^{[k]}$.*

Proof. Consider a designable structure S and let $w = w_1 \cdots w_n$ be a design for S. We will show that $w^{[k]}$ is a design for $S^{[k]}$. Let the i-th k positions in S be called the *region* i. Note that the positions in region i of $S^{[k]}$ correspond to the i-th position in S.

First, we will show that $S^{[k]}$ is an MFE structure for $w^{[k]}$. Consider an MFE structure S' of $w^{[k]}$. Define an *interaction graph* of S', denoted by $I(S') = (V_{I(S')}, E_{I(S')})$, as follows: the vertex set $V_{I(S')}$ is the set of positions in w, i.e., $\{1,\ldots,n\}$, and there is an edge between i and j in $I(S')$ if there exists a pair between a position in region i and a position in region j in S'. Note that $I(S')$ is a bipartite graph: indeed, vertices of any cycle in $I(S')$ are positions in w that alternate between A and U, or between C and G. Also note that $I(S')$ is an outer-planar graph: base pairs are pairwise non-crossing and can therefore be drawn without crossings on the upper half-plane, leaving the lower half-plane on the outer face. Assign each edge e in $E_{I(S')}$ a weight $c(e)$ equal to the number of pairs between regions i and j in S'. Note that the sum of all weights in $I(S')$, denoted as $\|E_{I(S')}\|$, equals $|S'|$. We have the following claim.

Claim 6.1. *If M is a maximum matching in $I(S')$ then $|S'| \leq k|M|$. Moreover, if $|S'| = k|M|$ then every minimum vertex cover of $I(S')$ covers every edge exactly once.*

Proof. Note that for any vertex i in $V_{I(S')}$, the sum of the weights of edges incident with i is at most k. Consider a smallest vertex cover C of $I(S')$, and take the sum of these inequalities over all

vertices i in the cover C:

$$\sum_{i \in C} \sum_{e \text{ incident with } i} c(e) \leq k|C|\,. \tag{1}$$

Since C is a vertex cover, the weight of every edge in $E_{I(S')}$ appears at least once on the left side of (1), hence $|S'| = \|E_{I(S')}\| \leq k|C|$. By König's Theorem, the maximum matching M in $I(S')$ has the same number of edges as C, i.e., $|S'| \leq k|M|$. The equality implies that the weight of every edge in $E_{I(S')}$ appears exactly once on the left side of (1), i.e., that vertex cover C covers every edge exactly once. □

Given a matching M in $I(S')$, we can construct a structure S_M for w with $|M|$ pairs as follows: for every edge $\{i, j\}$ in M, add pair (i, j). This is a valid (pseudoknot-free) structure, since M is a subgraph of outer-planar graph $I(S')$. It follows that $|M| \leq |S|$. If M is a maximum matching on $I(S')$, we have by Claim 6.1 that $|S'| \leq k|M| \leq k|S| = |S^{[k]}|$i.e., $S^{[k]}$ is an MFE structure for $w^{[k]}$. It also follows that $|S'| = k|M|$ and that $|M| = |S|$. Since S is a unique structure for w and $|S_M| = |M| = |S|$, we have that $S_M = S$, i.e., there is only one maximum matching in $I(S')$. We need the following claim to show that all connected components in $I(S')$ have at most 2 vertices.

Claim 6.2. *Let G be a connected bipartite graph on at least three vertices with unique maximum matching M. Then there exists a minimum vertex cover of G that covers some edge twice.*

Proof. First, we will show that every vertex in G is incident to an edge in matching M. Assume the contrary and consider all vertices in G which are incident to only non-matching edges. If two of these vertices are incident then the matching is not maximal. Otherwise, let u be such a vertex and v its neighbor. Vertex v must be incident to a matching edge. We can construct a new matching by removing this edge and adding edge uv, which contradicts the assumption that M is a unique maximal matching.

Take a maximal path P alternating between matching and non-matching edges in G. Let u be an endpoint of P and e the edge on P incident to u. If e is a non-matching edge then u must be incident to a matching edge, say f. By maximality of P, the other endpoint v of f must be on P. Since every internal vertex of P is incident to a matching on P, v must be the other endpoint of P and the edge incident to v on P must be a non-matching edge. Hence, we have an alternating cycle $P + f$ which contradicts the uniqueness of the maximal matching. Thus, P starts and ends with matching edges. Next, we show that u is a pendant vertex (has degree one). Assume to the contrary u is incident to another (non-matching) edge $f = uv$. By maximality of P, v is on P, which yields a cycle. If this cycle is even, we have an alternating cycle, which contradicts the uniqueness of the matching, and if it is odd, we have a contradiction with the fact that G is bipartite. Hence, both endpoints of P are pendant.

Consider a minimum vertex cover C of G. By well-known König's theorem, every minimum vertex cover in a bipartite graph uses exactly one endpoint of every edge of a maximum matching and no other vertices. Since the endpoints of P are pendant, and G is connected and has ≥ 3 vertices, P must have at least three edges. Since endpoints of P are pendant and incident to matching edges, we can assume that C does not contain endpoints of P, i.e, contains the second and last by one vertex of P. It is easy to see that at least one non-matching edge is covered twice by C. □

Consider a connected component K of $I(S')$. Since $I(S')$ has a unique maximum matching, so does K. If K has more than two vertices, it contains a minimum vertex cover of K that covers some edge twice. It follows that there is a minimum vertex cover of $I(S')$ that covers some edge twice. Hence, by Claim 6.1, $|S'| \leq k|M|$, a contradiction. It follows that every connected component of $I(S')$ has at most two vertices, hence, either S' is not MFE or $S' = S^{[k]}$. □

Theorem 7 (Result R8). *Each structure S without m_5 and $m_{3\circ}$ can be transformed into a $\Sigma_{2,0}$-designable structure S' by inflating a subset of its base pairs (at most one per band). Furthermore, this transformation can be done in $\Theta(n)$ time.*

Proof. We start with the greedy coloring of T_S. Since S does not contain m_5 and $m_{3\circ}$, it is a proper coloring and there is no node having both a grey child and an unpaired child. We will insert base pairs within S so that the grey nodes and any unpaired node end up at levels of different parities. If the root has a grey child, assign even parity to the grey nodes, otherwise (if the root has an unpaired child, or no grey and no unpaired children), assign even parity to the unpaired nodes.

Now we proceed from the children of the root towards leaves adjusting parity level for grey and unpaired nodes to keep one type even and the other one odd. We repeatedly apply the following simple operation on T_S: If the node N does not match its intended parity level. Denote N_P the parent of N (N_P is not the root as all children of the root already have the correct parity level) and N_{PP} the parent of N_P. Insert a new paired node N_N between N_{PP} and N_P, assign it with the color of N_P, and apply the greedy algorithm on N_N. Observe that N_P always takes either black or white color changing the parity level of all its descendants (including N). Note that the children of N_P may get recolored, we can even get one more grey child but after this operation the parity levels of all children of N are correct and we do not change parity levels outside the subtree rooted at N. After fixing all nodes, we get a separated proper coloring (which is actually the greedy coloring) of $T_{S'}$. Hence, by Theorem 5, S' is designable. Figure 4 illustrates this process. □

5 Conclusion, discussion and perspectives

In this work, we introduced the *Combinatorial RNA Design problem*, a *minimal* instance of the RNA design problem which aims at finding a sequence that admits the target structure as its unique base pair maximizing structure. First, we provided complete characterizations for the structures that can be designed using restricted alphabets. Then we considered the RNA design under a four-letter alphabet, and provide a complete characterization of designable saturated structures, i.e., free of unpaired positions. Turning to those target structures that contain unpaired positions, we provided partial characterizations for classes of designable/undesignable structures, and showed that the set of designable structures is closed under the stutter operation. Finally, we introduced structure-approximating version of the problem and, assuming that the input structure avoids two motifs, provided a structure approximating algorithm of ratio 2 for general structures.

An important question that is left open by this work is the computational complexity of the RNA design problem. Schnall-Levin *et al.* [17] established the NP-hardness of a more general problem, called the inverse Viterbi algorithm, which takes as input a stochastic grammar (representing the energy model) and a targeted parse tree (representing the structure), and outputs a sequence (design) whose most probable parsing should match the target. However this result does not settle the complexity of the RNA design, essentially because the proposed reduction relies critically on an encoding of 3-SAT instances within the input grammar. While the hypothetical *perfect* grammar/energy model for RNA folding probably differs from the currently accepted Turner model, it should ultimately reflect the laws of physics and should certainly not depend on the instance. As the reduction [17] requires a different grammar (i.e., energy model) for each instance, it does not seem easily adaptable into a proof that holds for a fixed energy model. Consequently, despite two decades of work on the subject, the computational tractability of RNA design is still open, either in its general instance and in our combinatorial version.

Besides complexity issues, natural extensions of this work may include the consideration of more general base pairing models, more realistic energy models (ideally, the Turner energy model [20]),

or the design under other objectives, such as the Boltzmann probability [22]. However, even the simplest of modifications, allowing $\mathsf{G}-\mathsf{U}$ base pairs, would invalidate parity properties that are critical to the proofs of some of our results and algorithms. More precise bounds for the ratio of the structure-approximating could be established. Finally, better algorithms could be designed for the problem, attempting to minimize the number of modifications so that a given structure becomes designable (or, more modestly, belongs to an identified class of designable structures).

References

1. R. Aguirre-Hernández, H. H. Hoos, and A. Condon. Computational RNA secondary structure design: empirical complexity and improved methods. *BMC Bioinformatics*, 8:34, 2007.
2. A. Avihoo, A. Churkin, and D. Barash. RNAexinv: An Extended Inverse RNA Folding from Shape and Physical Attributes to Sequences. *BMC Bioinformatics*, 12(1):319, Aug. 2011.
3. A. Busch and R. Backofen. INFO-RNA–a fast approach to inverse RNA folding. *Bioinformatics*, 22(15):1823–31, Aug 2006.
4. D. C. Dai, H. H. Tsang, and K. C. Wiese. RNADesign: Local Search for RNA Secondary Structure Design. In *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, 2009.
5. A. Esmaili-Taheri, M. Ganjtabesh, and M. Mohammad-Noori. Evolutionary solution for the RNA design problem. *Bioinformatics*, 30(9):1250–1258, May 2014.
6. Y. Frid and D. Gusfield. A simple, practical and complete o(n3/log n)-time algorithm for rna folding using the four-russians speedup. *Algorithms Mol Biol*, 5:13, 2010.
7. J. A. Garcia-Martin, P. Clote, and I. Dotu. RNAiFOLD: a constraint programming algorithm for RNA inverse folding and molecular design. *J Bioinform Comput Biol*, 11(2):1350001, Apr 2013.
8. S. Griffiths-Jones, A. Bateman, M. Marshall, A. Khanna, and S. R. Eddy. RFAM: an RNA family database. *Nucleic Acids Res*, 31(1):439–441, 2003.
9. I. L. Hofacker, W. Fontana, P. Stadler, L. Bonhoeffer, M. Tacker, and P. Schuster. Fast folding and comparison of RNA secondary structures. *Monatshefte für Chemie / Chemical Monthly*, 125(2):167–188, 1994.
10. C. Höner Zu Siederdissen, S. Hammer, I. Abfalter, I. L. Hofacker, C. Flamm, and P. F. Stadler. Computational design of RNAs with complex energy landscapes. *Biopolymers*, 99(12):1124–1136, Dec 2013.
11. A. Levin, M. Lis, Y. Ponty, C. W. O'Donnell, S. Devadas, B. Berger, and J. Waldispühl. A global sampling approach to designing and reengineering RNA secondary structures. *Nuc Acids Res*, 40(20):10041–52, Nov 2012.
12. R. B. Lyngsø, J. W. Anderson, E. Sizikova, A. Badugu, T. Hyland, and J. Hein. FRNAkenstein: multiple target inverse RNA folding. *BMC Bioinformatics*, 13:260, 2012.
13. D. H. Mathews, J. Sabina, M. Zuker, and D. H. Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J Mol Biol*, 288(5):911–940, May 1999.
14. R. Nussinov and A. Jacobson. Fast algorithm for predicting the secondary structure of single-stranded RNA. *Proc Natl Acad Sci U S A*, 77:6903–13, 1980.
15. V. Reinharz, Y. Ponty, and J. Waldispühl. A weighted sampling algorithm for the design of RNA sequences with targeted secondary structure and nucleotide distribution. *Bioinformatics*, 29(13):i308–i315, Jul 2013.
16. G. Rodrigo, T. E. Landrain, E. Majer, J.-A. Daròs, and A. Jaramillo. Full design automation of multi-state RNA devices to program gene expression using energy-based optimization. *PLoS Comput Biol*, 9(8):e1003172, 08 2013.
17. M. Schnall-Levin, L. Chindelevitch, and B. Berger. Inverting the Viterbi algorithm: an abstract framework for structure design. In *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, pages 904–911, 2008.
18. M. K. Takahashi and J. B. Lucks. A modular strategy for engineering orthogonal chimeric RNA transcription regulators. *Nucleic Acids Res*, 41(15):7577–7588, 2013.
19. A. Taneda. MODENA: a multi-objective RNA inverse folding. *Adv Appl Bioinform Chem*, 4:1–12, 2011.
20. D. H. Turner and D. H. Mathews. NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Res*, 38(Database issue):D280–D282, Jan 2010.
21. S. Y. Wu, G. Lopez-Berestein, G. A. Calin, and A. K. Sood. RNAi therapies: Drugging the undruggable. *Science Translational Medicine*, 6(240):240ps7, 2014.
22. J. N. Zadeh, B. R. Wolfe, and N. A. Pierce. Nucleic acid sequence design via efficient ensemble defect optimization. *J Comput Chem*, 32(3):439–52, Feb 2011.
23. Y. Zhou, Y. Ponty, S. Vialette, J. Waldispuhl, Y. Zhang, and A. Denise. Flexible RNA Design Under Structure and Sequence Constraints Using Formal Languages. In *Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics*, BCB'13, pages 229–238. ACM, 2013.
24. M. Zuker and P. Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Res*, 9:133–148, 1981.