

An Efficient Algorithm to Compute the Minimum Free Energy of Interacting Nucleic Acid Strands

Ahmed Shalaby ✉️🏠^{ID}

Hamilton Institute and Department of Computer Science, Maynooth University, Ireland

Damien Woods ✉️🏠^{ID}

Hamilton Institute and Department of Computer Science, Maynooth University, Ireland

Abstract

The information-encoding molecules RNA and DNA bind via base pairing to form an exponentially large set of *secondary structures*. Practitioners need algorithms to predict the most favoured structures, called minimum free energy (MFE) structures, or to compute a partition function that allows assigning a probability to any structure. MFE prediction is NP-hard in the presence pseudoknots – base pairings that violate a restricted planarity condition. However, for single-stranded unspseudoknotted structures, there are polynomial time dynamic programming algorithms. For multiple strands, the problem is significantly more complicated: Codon, Hajiaghayi and Thachuk [DNA27, 2021] proved it NP-hard for N bases and $\mathcal{O}(N)$ strands. Dirks, Bois, Schaeffer, Winfree and Pierce [SIAM Review, 2007] gave a polynomial time partition function algorithm for multiple ($\mathcal{O}(1)$) strands, now widely-used, however their technique did not generalise to MFE which they left open.

We give an $\mathcal{O}(N^4)$ time algorithm for unspseudoknotted multiple ($\mathcal{O}(1)$) strand MFE prediction, answering the open problem from Dirks et al. The challenge lies in considering the rotational symmetry of secondary structures, a global feature not immediately amenable to local subproblem decomposition used in dynamic programming. Our proof has two main technical contributions: First, a characterisation of symmetric secondary structures implying only quadratically many need to be considered when computing the rotational symmetry penalty. Second, that bound is leveraged by a backtracking algorithm to efficiently find the MFE in an exponential space of contenders.

2012 ACM Subject Classification Theory of computation → Algorithm design techniques; Theory of computation → Dynamic programming

Keywords and phrases Minimum free energy, MFE, partition function, nucleic acid, DNA, RNA, secondary structure, computational complexity, algorithm analysis and design, dynamic programming

Digital Object Identifier 10.4230/LIPIcs.ICALP.2025.130

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: <https://arxiv.org/abs/2407.09676> [33]

Funding Supported by Science Foundation Ireland (SFI) under grant number 20/FFP-P/8843, European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No 772766, Active-DNA project), and Funded by the European Union - European Innovation Council (EIC) and SMEs Executive Agency (EISMEA). Grant number 101115422, DISCO project. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union, ERC, EIC, SMEs Executive Agency (EISMEA) or SFI. Neither the European Union nor the granting authority can be held responsible for them.

Acknowledgements We thank Constantine Evans for his helpful comments on the origin of the MFE rotational symmetry penalty from statistical mechanics, Mark Fornace, Niles Pierce, Dave Doty, Erik Winfree and Anne Condon for helpful comments.



© Ahmed Shalaby and Damien Woods;

licensed under Creative Commons License CC-BY 4.0

52nd International Colloquium on Automata, Languages, and Programming (ICALP 2025).

Editors: Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis

Article No. 130; pp. 130:1–130:20



Leibniz International Proceedings in Informatics

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany



■ **Table 1** Run-time bounds for algorithms that compute MFE and partition function for unpseudoknotted² nucleic acid systems, and a hardness result. N is the total number of bases of all strand(s) in the system, i.e. the sum of all strand lengths. Results are shown for input being a single strand, or multiple strands bounded by a constant or unbounded/growing with N . Results are proven in the nearest neighbour energy model studied here, except the APX-hardness result which is proven in the maximum matching model [9], and still open for nearest neighbour model.

Input type	MFE	Partition function
Single strand	$\mathcal{O}(N^3)$ [47, 46, 25, 24, 39] ⁵	$\mathcal{O}(N^3)$ [23] ⁵
Multiple strands, bounded, i.e. $c = \mathcal{O}(1)$ strands	$\mathcal{O}(N^4(c-1)!)$ [Theorem 1]	$\mathcal{O}(N^3(c-1)!)$ [10] ⁵
Multiple strands, unbounded, i.e. $\mathcal{O}(N)$ strands	APX-hard [9]	Open problem

1 Introduction

The *primary structure* of a DNA strand is simply a word over the alphabet $\{A, C, G, T\}$, or $\{A, C, G, U\}$ for RNA. Bases may bond in pairs, A binds to T and C binds to G, and a set of such pairings for a strand is called a *secondary structure* as shown in Figure 1(a); typically each strand has exponentially many possible secondary structures. Mainly, what practitioners care about are probabilities of a given secondary structure or class of secondary structures. For that, each secondary structure S has an associated, typically negative, real valued *free energy* $\Delta G(S)$, where more negative is deemed more favourable. Thus the most favourable is the secondary structure, or structures, with minimum free energy (MFE). More generally, the Boltzman distribution is a probability distribution on secondary structures at equilibrium: the probability of S is $p(S) = \frac{1}{Z} e^{-\Delta G(S)/k_B T}$, where Z is a normalisation factor called the partition function:

$$Z = \sum_{S \in \Omega} e^{-\Delta G(S)/k_B T} \quad (1)$$

that is, an exponentially weighted sum of the free energies over the set Ω of all secondary structures, where k_B is Boltzmann's constant and T is temperature in Kelvin.¹

1.1 Related work

Decades ago, the deep relationship between secondary structures and dynamic programming algorithms was established [47, 46, 25, 24, 39, 23]. If a secondary structure can be drawn as a polymer graph without edge crossings it is called unpseudoknotted (Figure 1(c)). The earliest polynomial time algorithms were for single-stranded unpseudoknotted secondary structures, where the absence of crossings allows for planar decompositions of secondary structures suited to dynamic programming techniques.² For a single RNA/DNA strand, both

¹ Although we use a few terms from physics/chemistry, the objects we analyse in this paper are sets of tuples (i.e. secondary structures) with straightforward mathematical definitions amenable to dynamic programming and mathematical analysis. In particular, we ignore 3D geometry—the literature has already established that simple secondary structure models nicely thread the line between having a mathematical/computational model, and being predictive for experimental conditions [10, 17, 30].

² We do not study pseudoknotted structures here, indeed most literature ignores pseudoknots for both modelling and algorithmic considerations: Energy models for pseudoknots are difficult to formulate for geometric reasons [10]. Pseudoknotted MFE prediction is NP-hard even for a single strand [1, 20, 21].

MFE and partition function are computable in $\mathcal{O}(N^3)$ time⁵ (Table 1), using the standard nearest neighbour energy model³ that will be formally defined in Section 2.

Work in DNA computing [42, 26, 35, 38, 14, 6, 32, 44], and nucleic acid nanotechnology more generally [16, 41], involves building molecular systems and structures with, to date, hundreds, and soon, thousands, of interacting strands, so there is a need for better algorithms for these multi-stranded “inverse design problems” [8, 13]. And, of course, biologists need to understand molecular structure in order to understand and predict molecular interactions. However, when there are multiple interacting strands, the situation becomes significantly more complicated than the single-stranded case for two reasons: First, for a secondary structure to be unspseudoknotted, it implies there should be *at least one* permutation of the strands without crossings on the polymer graph [10] (Figure 1). Second, if strand types are repeated then so-called *rotational symmetries* (Figure 2) arise that need to be accounted for in the model to match the underlying statistical mechanics⁴, otherwise structures may be over- or undercounted, leading to incorrect probabilities in the Boltzmann distribution, in other words: incorrect predicted free energy of a secondary structure.

For multiple strands, albeit a constant number $c = \mathcal{O}(1)$, Dirks, Bois, Schaeffer, Winfree and Pierce [10] gave a polynomial time partition function algorithm that runs in time $\mathcal{O}(N^3(c-1)!)$.⁵ Interestingly, Boehmer, Berkemer, Will, and Ponty [3] recently reduced the c -strand parameterized running time for computing partition function and *symmetry-naïve MFE* (i.e. ignoring rotational symmetry) from factorial, $\mathcal{O}(N^3(c-1)!)$, to exponential, $\mathcal{O}(N^3 3^c)$. In future work, it could be possible to similarly improve the $(c-1)!$ factor in our MFE algorithm via this result.

In the case where the number of strands is non-constant, in particular $c = \mathcal{O}(N)$, Codon, Hajiaghayi and Thachuk [9] showed MFE is NP-hard, and even APX-hard.⁶ For the second, rotational symmetry, problem, in order to compute partition function, Dirks et al. [10] found an algebraic link between the overcounting and the rotational symmetry correction problems, which allowed both to be solved simultaneously, aided by the exponential nature of the partition function. Surprisingly, that trick does not work for MFE: Since MFE prediction is minimization-based, there is no secondary structure overcounting problem in MFE prediction. In other words, unlike the case of summation-based partition function, algorithmic examination of repeated secondary structures will not change the outcome of energy minimization. Hence, the absence of the overcounting problem makes MFE prediction harder to solve, and was left open by Dirks et al. [10]. For the special case of $c = 2$ strands, Hofacker, Reidys, and Stadler [17] gave an $\mathcal{O}(N^6)$ algorithm.

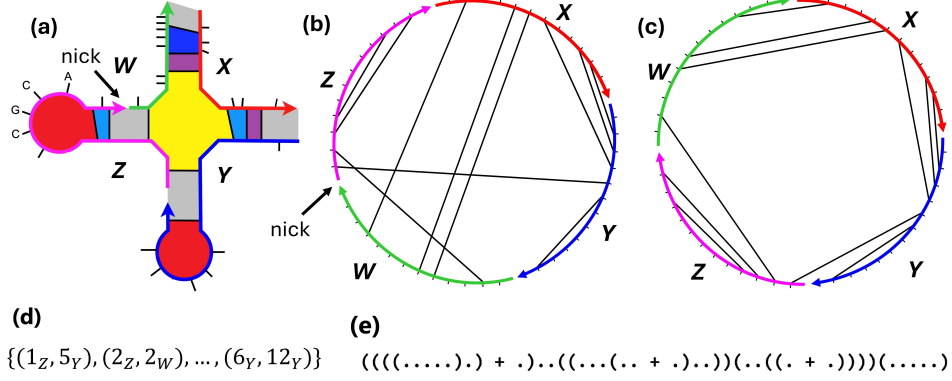
Early NP-hardness results [1, 20] used a simple energy model (no loops, only consecutive base pairs forming “stacks” contribute to free energy). Already seeing hardness results with simple energy models make it unlikely that MFE prediction problem for more realistic energy models [9] is tractable. Nevertheless, dynamic programming algorithms exist for restricted classes of pseudoknots, for both MFE [28, 37, 7, 18, 27] and partition function [11, 12].

³ The standard model we use is variably called the nearest neighbour model, the Turner model, or loop energy model. Versions of the model have been implemented in software suites such as NUPACK [10, 12, 15], ViennaRNA [19] and mfold [45], for both RNA and DNA [29, 30].

⁴ This fact from statistical mechanics is discussed in some papers [10, 17], although we’ve not found its full derivation in the modern nucleic-acid algorithmic literature. We leave a first-principles derivation for future work.

⁵ We note that in the original literature [23, 10] that established these results, the polynomial is to the power 4 due to interior loop contributions (i.e. $\mathcal{O}(N^4)$ for single-stranded and $\mathcal{O}(N^4(c-1)!)$ for multistranded). The subsequent improvement in the exponent from 4 to 3 is achieved using techniques in Dirks and Pierce (2003) [11] to handle the energy contributions of interior loops more efficiently in dynamic programming.

⁶ This hardness result holds whether or not rotational symmetries are accounted for in the energy model.



■ **Figure 1** A DNA (or RNA) secondary structure S with $c = 4$ strands and two of its $(c - 1)! = 6$ polymer graphs. (a) One of the many possible secondary structures for four DNA strands W, X, Y, Z . Short black lines represent DNA bases (a few are shown ... C, G, C, A ...), and long lines represent base pairs (drawing not to scale). Loops are colour-coded as follows: stack (purple), multiloop (yellow), hairpin (red), bulge (light blue), internal (dark blue), external (grey). Black arrow: the small gap between two strands is called a *nick*. (b) Polymer graph for the strand ordering $\pi' = WZXY$, denoted $\text{Poly}(S, \pi')$, showing base-pair crossings. (c) By reordering to $\pi = ZWXY$ we get another polymer graph $\text{Poly}(S, \pi)$ for S , without crossings, hence S is unpsudoknotted. (d) The secondary structure S written mathematically as a set of base pairs of indices, i.e. in the format of Definition 2. (e) Since S is unpsudoknotted, it can also be written in dot-parens-plus notation [43].

1.2 Statement of main result

We give an efficient solution to the $\mathcal{O}(1)$ -strand MFE problem, the first that runs in polynomial time. Our main result is stated as the following theorem, whose proof is in Section 5:

► **Theorem 1.** *There is an $\mathcal{O}(N^4(c - 1)!)$ time and $\mathcal{O}(N^4)$ space algorithm for the Minimum Free Energy unpsudoknotted secondary structure prediction problem, including rotational symmetry, for a set of $c = \mathcal{O}(1)$ DNA or RNA strands of total length N bases.*

In Section 5 we give a time-space trade-off for our result, by showing a variation of the algorithm runs in $\mathcal{O}((N^4 \log N)(c - 1)!)$ time but $\mathcal{O}(N^3)$ space.

We use the standard [10] definition of free energy (Equation (2)) of multistranded unpsudoknotted secondary structures, which includes rotational symmetry, see Section 2 for formal definitions. We first give an extensive overview of the proof and paper structure, followed by future work.

1.3 Proof overview and paper structure

1.3.1 The main challenge: handling rotational symmetry

Typically, although not always, each DNA base pair that forms represents a decrease in free energy (more favourable). In a multi-stranded system, when several strands bind together the entropy of the overall system is decreased since there are now less system states due to their being less free molecules. Thus the energy model for multistranded systems includes an entropic *association penalty* for every extra strand, beyond the first, bound into a multistranded molecular complex [10] (typically positive, less favourable). However, statistical mechanics tells us to be careful about symmetry: with multiple identical strands in a complex it is possible that the complex is rotationally symmetric, intuitively there

are several complexes, identical up to rotation of their polymer graphs (Figure 2).⁷ These so-called indistinguishable complexes, in turn imply that another (positive) penalty should be applied to account for the difference in entropy between a similar, but distinguishable, complex without rotational symmetry [4, 2, 34, 15].⁴ Section 2 gives definitions to formalise these concepts, including: DNA, unpseudoknotted secondary structure, polymer graph, free energy including rotational symmetry (Equation (2)) and MFE (Equation (3)). In particular, Section 2.3 gives a group-theoretic definition of rotational symmetry, to help formalise some of the prior work.

1.3.2 General approach to find the true MFE

One obvious idea might be to find a dynamic programming algorithm that directly handles rotational symmetry. However, this approach suffers from rotational symmetry being a *global property* of an entire system state (secondary structure), whereas dynamic programming relies on piecing together subproblems that are individually unaware of the global context – or more precisely may be used in multiple global contexts whether symmetric or not.

Instead, our strategy is to first compute what we call the *symmetry-naïve MFE* (snMFE) that (incorrectly) assumes all strands are distinct and thus does not compute correct free energies for rotational symmetries. We use the snMFE algorithm of Dirks et al. [10], that assumes all strands are distinct, but augmented to return extra dynamic programming matrices (Algorithm 1 in Appendix B of the full version [33]). We prove that we can use these extra matrices to help reduce the time complexity of the backtracking algorithm, hence compute the required symmetry correction to the snMFE value efficiently, as follows.

1.3.3 Polynomial upper bound: intuition for Section 3

Our goal is to show that, after running our augmentation of the known algorithm for snMFE, we have implicit access to a collection of secondary structures that are “not too far” from the true MFE – where by “not too far” we mean we have a polynomial upper bound on the number of structures to be considered by a fast backtracking algorithm that finds the true MFE structure.

First, to see how we find this polynomial bound, imagine the augmented snMFE algorithm finds that the secondary structure with snMFE is rotationally *asymmetric*, hence we are done, we know that the snMFE value is in fact the true MFE. Otherwise, we have a *rotationally symmetric* secondary structure: ideally we would like to compute its rotational symmetry degree R (takes linear time in the size of the secondary structure), and then return $\text{snMFE} + k_B T \log R$ as the true MFE, but this approach is doomed to fail since there could also be structures with lower true MFE, i.e. in the real interval $[\text{snMFE}, \text{snMFE} + k_B T \log R) \subset \mathbb{R}$.

Leveraging the two properties of being (a) unpseudoknotted and (b) rotationally symmetric, in Section 3 we define a class of cuts of a structure’s polymer graph (Figures 1 and 3) that we call *pizza cuts*, or, more formally, *admissible symmetric backbone cuts* (Definition 15). These cuts are radially symmetric, hence the name pizza cut – how one slices a pizza from disk-edge to centre. In Lemmas 14 and 24, we show that there are at most a polynomial number of pizza cuts that symmetric structures may have. We use the pizza metaphor to denote secondary structure, and pizza slice for substructure.

⁷ Formally, we mean the permutation representing the complex is rotationally symmetric.

Then, when we do a backtracking-based search (below), through the dynamic programming matrices from the structure(s) with snMFE, to larger free energies: if we find two different symmetric pizzas, but with the same pizza cuts, we make a new pizza, by swapping a slice from one with a slice from the other. We prove that the new pizza is (a) guaranteed to be asymmetric and (b) has free energy sandwiched between the snMFE values of two symmetric structures (Lemmas 22 and 23). Hence, during the backtracking and in the worst case of reaching the polynomial upper bound exhausting the set of all *admissible symmetric backbone cuts*, it is guaranteed to output the true MFE structure either it was symmetric or not.

1.3.4 Backtracking to find the true MFE

It remains to show how we will do the backtracking search mentioned above. In Section 4 of the full version of this paper [33], we analyse the backtracking algorithm from Appendix C of [33], which is a polynomial time algorithm over the exponentially large set of structures “close” to the true MFE value. It scans all secondary structures within an energy level starting with the snMFE energy level, and goes on to sequentially scan higher levels in low-to-high order. The scanning process at any energy level \mathcal{E} guarantees that each secondary structure belonging to \mathcal{E} should be scanned exactly once.

The backtracking algorithm runs until one of the following conditions occurs: (1) It scans an asymmetric secondary structure S , or (2) it exceeds the polynomial upper bound \mathcal{U} of the number of symmetric secondary structures (i.e. the number of distinct pizza cuts) to be scanned, or (3) the backtracking will start scanning a new energy level $\mathcal{E}' > \mathcal{B}$, where \mathcal{B} is the current best candidate for MFE (the starting value for \mathcal{B} is $\mathcal{B} = \text{snMFE} + k_B \log v(\pi)$ where $v(\pi)$ is the highest degree of rotational symmetry, Definition 7). Then, based on the condition that will occur, the algorithm directly returns the true MFE, and a secondary structure which has the true MFE will also be constructed. The short proof of Theorem 1 in Section 5 ties these results together to give the final analysis of our main result. Full technical details of the backtracking algorithm are in Section 4, and Algorithm 2 in Appendix C in the full version of this work [33].

1.4 Future work

Our algorithm runs in polynomial time $\mathcal{O}(N^4(c-1)!)$ for the case of $c = \mathcal{O}(1)$ strands, the $(c-1)!$ term coming from the fact that our algorithm, as well as the snMFE algorithm from Dirks et al. [10], is assumed to be called from an outer loop that explicitly tries all $(c-1)!$ cyclic strand permutations. Can we increase the number of strands and still have a polynomial time algorithm? We know “not by much”, since the problem is NP-complete when $c = \mathcal{O}(N)$ in a simpler energy model that merely maximises the count of base pairs [9].

Our MFE algorithm exploits a polynomial upper bound, \mathcal{U} , on the number of so-called symmetric secondary structures, or distinct pizza cuts. That bound is linear in “most” cases (Lemma 26), but quadratic in one special subcase (Lemma 24) of 2-fold rotational symmetry with a central internal loop. Reducing that special case to linear would subtract one from our algorithm’s running time and space exponent.

The computational complexity of partition function for DNA/RNA strands is less well understood than MFE. Table 1 shows an open problem for partition function on multiple strands. Intuitively, it seems that partition function should be at least as hard as MFE, however that intuition is tempered by the fact that Dirks et al.’s approach found an efficient algorithm for partition function but not MFE. Nevertheless: are there settings where partition function, or problems counting numbers of structures, are #P-hard?

2 Definition of multi-stranded DNA systems and basic lemmas

Intuitively, a single DNA strand s is a sequence of nucleotide bases connected by covalent bonds which together make up the backbone of s , with the left end of the sequence corresponding to the 5' end of s and the right end corresponding to the 3' end. When drawing s we label the 3' end with an arrow which also shows the strand directionality, see Figure 1. Hydrogen bonds can form between Watson-Crick base pairs, namely C–G and A–T.

Formally, a DNA strand s is a word over the alphabet of DNA bases $\{A, T, G, C\}$, indexed from 1 to $|s|$, where $|s|$ denotes the length of s . A base pair is a tuple (i, j) such that $i < j$. For any c strands, we will assign to each of them a unique distinct identifier in $\{1, \dots, c\}$ [10]. Each base is specified by a strand identifier and a position on that strand, i_s denotes the base of index i of strand s .

2.1 Connected unpsudoknotted secondary structures and polymer graphs

► **Definition 2** (Secondary structure S). *For any set of c DNA strands, a secondary structure S is a set of base pairs such that each base appears in at most one pair, i.e. if $(i_n, j_m) \in S$ and $(k_q, l_r) \in S$ then i_n, j_m, k_q, l_r are all distinct.*

The graph representation of a secondary structure S is the graph $G = (V, E)$, where V is the set of bases of each strand $s \in \{1, \dots, c\}$, and $E = E_v \cup E_b$, where E_v is the set of *covalent backbone bonds* connecting base i_n with base $(i+1)_n$ for all bases $i = 1, 2, \dots, |n| - 1$ on all strands $n \in \{1, \dots, c\}$, and $E_b = S$ is the set of base pairs in S . E_v and E_b are disjoint.

The set of circular permutations, Π , of c strands has size $(c-1)!$ [5] (e.g., for the three strands $\{A, B, C\}$, $\Pi = \{ABC, ACB\}$), e.g., the orderings ABC , BCA , and CAB are the same on a circle. Next, we define a polymer graph for each π , see also Figure 1.

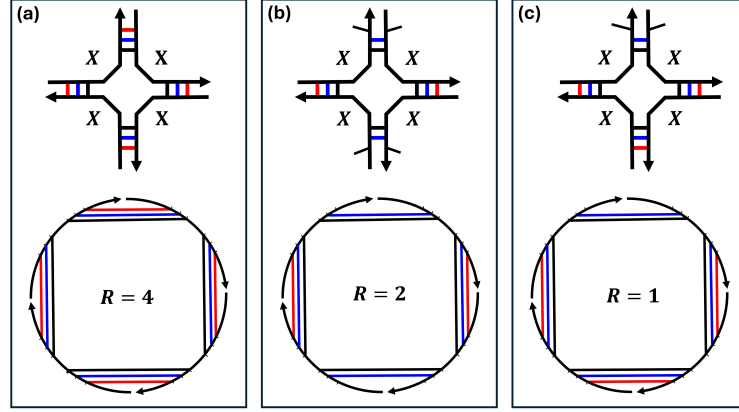
► **Definition 3** (Polymer graph). *For any secondary structure S , and any ordering π of its c strands, the polymer graph representation of S , denoted $\text{Poly}(S, \pi)$, is a graph representation of S , embedded in the unit disk from \mathbb{R}^2 , where the c strands are placed in succession from their 5' to 3' ends around the circumference of the circle in the order given by π , the bases, V , are spaced evenly around the circle circumference, each element of E_v is represented by an arc on the circumference between covalently-bonded bases, and each element of E_b is represented by a chord between two different bases.*

► **Definition 4** (Unpsudoknotted secondary structure). *A secondary structure S is unpsudoknotted if there exists at least one circular permutation $\pi \in \Pi$ such that $\text{Poly}(S, \pi)$ is planar (none of the chords cross), otherwise S is pseudoknotted. An example is shown in Figure 1.*

► **Remark 5.** In the rest of the paper we use N to denote the total number of bases of a secondary structure S . A secondary structure S is connected if the graph representation of S is a connected graph. In this work, we are only interested in connected unpsudoknotted secondary structures.

2.2 Free energy of a secondary structure

In the nearest neighbour energy model [22, 30, 10], any connected unpsudoknotted secondary structure S can be decomposed into different loop types [36, 30, 22]: namely hairpin loops, interior loops, exterior loops, stacks, bulges, and multiloops as shown in Figure 1. As usual,



■ **Figure 2** Three secondary structures with their associated polymer graphs. In each case, there is a single complex with four identical (indistinguishable) strands of strands of type X , but with different symmetry degree R . (a) Symmetry degree $R = 4$ (rotation by 90° gives the same secondary structure). (b) Symmetry degree $R = 2$ (rotation by 180° gives the same secondary structure). (c) Symmetry degree $R = 1$ (asymmetric secondary structure).

let k_B be Boltzmann's constant and T is the temperature in Kelvin (also a constant).⁸ The free energy of S is defined as the sum of three terms:⁹

$$\Delta G(S) = \sum_{l \in S} \Delta G(l) + (c - 1) \Delta G^{\text{assoc}} + k_B T \log R. \quad (2)$$

- the first is itself the sum of the (well-defined, empirically-obtained) free energies $\Delta G(l)$ of S 's constituent loops [10], where the energy of each loop $l \in S$, is defined with respect to the free energy of the unpaired reference state.
- ΔG^{assoc} is the entropic association [10] penalty applied for each of the $c - 1$ strands added to the first strand to form a complex of c strands.
- R is the rotational symmetry of the secondary structure S , illustrated in Figure 2, and to be formally defined in Section 2.3. In particular, since favourable free energies are usually negative, the term $k_B T \log R \geq 0$ corresponds to the reduction in the entropic contribution of S , as any secondary structure with an R -fold rotational symmetry has a corresponding R -fold reduction in its distinguishable conformational space [10] as shown in Figure 2.¹⁰ Dynamic programming algorithms to date for multi-stranded MFE ignored this term for reasons we outline in Section 2.3.

For c strands, we let Ω be the set (usually called the *ensemble*) of all connected unpseudoknotted secondary structures. For any circular permutation $\pi \in \Pi$ of the c strands, let $\Omega(\pi) \subseteq \Omega$ be the subset of Ω such that each connected unpseudoknotted secondary structure $S \in \Omega(\pi)$ is representable as a crossing-free polymer graph with circular permutation π .

► **Remark 6** (S , or $\text{Poly}(S, \pi)$). Dirks et al. [10] showed, in their representation theorem (Theorem 2.1), that the sets $\Omega(\pi)$, for all $\pi \in \Pi$, form a partitioning of Ω , which means that every connected unpseudoknotted secondary structure belongs to exactly one $\Omega(\pi)$ for

⁸ All results hold if we assume these are typical values from physics, or just 1 in appropriate units.

⁹ Throughout this paper $\log n = \log_e n$.

¹⁰ This is perhaps counter-intuitive. In a follow-up expanded version of this paper we will give a full statistical mechanics explanation, which uses the symmetry penalty to offset the fact that non-symmetrical structures are overcounted from algorithmic perspective.

some $\pi \in \Pi$. Hence to avoid the cumbersome phrase *c-strand connected unpseudoknotted secondary structure S with strand ordering π and polymer graph $\text{Poly}(S, \pi)$* we simply write S , or $\text{Poly}(S, \pi)$.

Predicting the minimum free energy means finding a minimum over the ensemble Ω . The known strategy is to deal with each partition $\Omega(\pi)$ separately, then finding their minimum:

$$\text{MFE} = \min_{S \in \Omega} \Delta G(S) = \min_{\pi \in \Pi} \left\{ \min_{S \in \Omega(\pi)} \Delta G(S) \right\}. \quad (3)$$

2.3 Definition of multi-stranded rotational symmetry

Here, we formalise rotational symmetry. In the previous section we assigned each one of the c strands a unique identifier, dealing with them as distinct strands even if two or more have the same sequence. But in most experimental settings, strands with the same sequences are *indistinguishable* in the sense that they behave identically with respect to relevant measurable quantities [10]. Mathematically, we say that *two strands are indistinguishable* if they have the same sequence. Also, two *secondary structures are indistinguishable* if there exists a permutation of the implied unique strand ordering (Remark 6), that maps indistinguishable strands onto each other while preserving all base pairs, otherwise, the two structures are distinct [10].

For any c strands, not necessarily distinct, they consist of $k \leq c$ *strand types*, usually denoted by uppercase English letters X, Y, \dots .¹¹ A *multi-stranded DNA system* $M = \{(t_1, n_1), (t_2, n_2), \dots, (t_k, n_k)\}$, is a *multiset* of k strand types t_1, \dots, t_k with repetition numbers $n_1, \dots, n_k \in \mathbb{N}$ such that $n_1 + \dots + n_k = c$.¹² For such a multiset M we can think of each circular permutation π as a string over strand types such that each strand type t_i appears exactly n_i times (e.g., $M = \{(X, 6), (Z, 3)\}$ one valid π is $XZXXZXZX$).

► **Definition 7** (Symmetry degree of a permutation). *For any circular permutation π , we say $n \in \mathbb{N}$ is a symmetry degree of π if $\pi = y^n$ for some y , a prefix of π .*

For example, $\{1, 2, 4\}$ are the symmetry degrees of $\pi = XZXXZXZX$ which follows from $\pi = (XZXXZXZX)^1 = (XZXX)^2 = (XZ)^4$.

For any circular permutation π , its maximum symmetry degree is denoted $v(\pi)$, and the corresponding repeating prefix x , such that $x^{v(\pi)} = \pi$, is the *fundamental component* of π . It can be seen that x is the smallest prefix that repeats over π . Indeed, $v(\pi)$ is the number of cyclic permutations that map each strand to a strand of the same type. Any repeating prefix of π must be a multiple of its fundamental component, see Lemma 33, Appendix A in the full version [33].

► **Remark 8** (Notation: X_m^n). For any circular permutation π , its *augmented version* gives full ordering information for each fundamental component. For example, if $\pi = XYXZXYXZ$, then its fundamental component is $XYXZ$, and $X_1^1 Y_1^1 X_2^1 Z_1^1 X_1^2 Y_1^2 X_2^2 Z_1^2$ is its augmented version, such that X_m^n , means the m th strand of type X in the n th fundamental component of π .

¹¹In contrast with some of the literature. we exclude using $\{A, T, G, C\}$ for strand types; to avoid any confusion between strand and base types.

¹²It is known [31] how to efficiently reduce the circular permutation space by getting rid of circular permutations that are redundant due to indistinguishable strands, which is important to consider when computing the partition function but needed for MFE.

We can visualize any ordering π by representing it as a *regular $v(\pi)$ -gon* with each of its $v(\pi)$ vertices representing a fundamental component. Let $\rho = (1\ 2\ 3\ \dots\ v(\pi))$,¹³ and consider the cyclic group¹⁴ G^π generated by ρ . Intuitively, G^π is the group of all $v(\pi)$ rotational motions in plane of the regular $v(\pi)$ -gon that give the same $v(\pi)$ -gon. We can represent G^π as follows: $G^\pi = \{\rho^0, \rho^1, \dots, \rho^{v(\pi)-1}\}$, where ρ^i represents rotation of the regular $v(\pi)$ -gon by the angle of $i \times \frac{360^\circ}{v(\pi)}$, where $|G^\pi| = v(\pi)$.

Now, we are ready to define the rotational symmetry of a secondary structure and a strand ordering π , intuitively, the number of rotations of its polymer graph that give the same polymer graph, as shown in Figure 2.

► **Definition 9** (*R-fold rotationally symmetric structure*). *A connected unpseudoknotted secondary structure S and strand ordering π (and thus polymer graph $\text{Poly}(S, \pi)$) is R -fold rotational symmetric, or simply rotationally symmetric, if for any base pair (i, j) in the polymer graph $\text{Poly}(S, \pi)$ the rotation of that base pair by multiples of $(360^\circ/R)$ is also in $\text{Poly}(S, \pi)$. More formally: $(i_{X_k^l}, j_{Y_m^n}) \in \text{Poly}(S, \pi)$, iff $(i_{X_k^{a(l)}}, j_{Y_m^{a(n)}}) \in \text{Poly}(S, \pi)$ for all $a \in H \leq G^\pi$, where H is the largest subgroup¹⁵ satisfying the condition, and if $|H| = R$.*

► **Remark 10.** In Def. 9, we restricted H to be the *largest* subgroup so as to be aligned with the entropic reduction penalty due to symmetry that appears in Equation (2), even if from a geometric perspective any R -fold rotationally symmetric secondary structure should be also R' -fold rotationally symmetric if R' divides R .

3 A polynomial upper bound on a class of rotationally symmetric secondary structures

► **Remark 11.** In this section, we assume a global indexing of all bases from 1 to N , and use square brackets, $[i, i+1]$, to denote the covalent bond connecting bases i and $i+1$, given that both belong to the same strand. This notation also helps to differentiate covalent bonds $[i, i+1]$ from base pair (j, k) (hydrogen bonds) notation.

► **Definition 12** (*R-symmetric backbone cut generated by a covalent bond*). *For a connected unpseudoknotted secondary structure S and strand ordering π , the R -symmetric backbone cut generated by covalent bond $b = [i_{A_m^n}, (i+1)_{A_m^n}]$ is $\mathcal{C}_R^b = \{[i_{A_m^{a(n)}}, (i+1)_{A_m^{a(n)}}] : \text{for all } a \in H \leq G^\pi \text{ such that } |H| = R\}$. We call b a symmetric backbone cut generator. Figure 3 shows an example.*

Only one covalent bond is needed to generate its corresponding R -symmetric backbone cut. Also, note that this definition excludes any cut through nicks by Remark 11. Lemma 14 in the following subsection shows that the number of unique symmetric backbone cuts is linear in N .

3.1 Linear upper bound on number of unique symmetric backbone cuts

The following lemma restricts us to deal with only specific and constant number of different folding rotational symmetries, and hence constant different symmetry corrections in total. The proof of this lemma is in Appendix A in the full version [33].

¹³ Here we use algebraic cycle notation. The order of ρ , denoted by $o(\rho)$, is the length of ρ which is $v(\pi)$.

¹⁴ G^π is isomorphic to $C_{v(\pi)}$, cyclic group of order $v(\pi)$.

¹⁵ $H \leq G$, notationally means H is a subgroup of G . Every subgroup of cyclic group is also cyclic.

► **Lemma 13.** *If S is R -fold rotationally symmetric secondary structure, with a specific circular permutation π , then R must be a divisor of $v(\pi)$.*

► **Lemma 14** (Upper bound on unique symmetric backbone cuts). *For any connected unpseudoknotted secondary structure S of $c = \mathcal{O}(1)$ strands with N total bases, with a specific strand ordering π , the number of unique symmetric backbone cuts is $\frac{N-c}{v(\pi)} [\sigma(v(\pi)) - v(\pi)] = \mathcal{O}(N)$, where $\sigma(v(\pi))$ is the sum of divisors of $v(\pi)$.*

Proof. In general, any covalent bond is a potential candidate for a symmetric backbone cut generator. For secondary structure S with N bases and c strands, there are $N - c$ covalent bonds in S by excluding all nicks. We need to compute the total number of all possible symmetric backbone cuts for every possible symmetry degree R . Given a specific R , the number of R -symmetric backbone cuts is $\frac{N-c}{R}$, since for any covalent bond x in a cut \mathcal{C}_R^b , we have $\mathcal{C}_R^x = \mathcal{C}_R^b$ (all bonds in that cut generate the same cut). And, hence for any two covalent bonds x and y , either $\mathcal{C}_R^x = \mathcal{C}_R^y$ or they are disjoint.

Because of symmetry (see Lemma 13), $R > 1$ and R divides $v(\pi)$ (denoted $(R \neq 1) | v(\pi)$ below). Assume that $d_1, d_2, \dots, v(\pi)$ are divisors of $v(\pi)$ such that $d_i \neq 1$, and since divisors happen in pairs ($d_i d'_i = v(\pi)$), then the total number of symmetric backbone cuts is

$$\begin{aligned} \sum_{(R \neq 1) | v(\pi)} \frac{N-c}{R} &= (N-c) \sum_{(R \neq 1) | v(\pi)} \frac{1}{R} \\ &= (N-c) \left[\frac{1}{d_1} + \frac{1}{d_2} + \dots + \frac{1}{v(\pi)} \right] \\ &= (N-c) \left[\frac{d'_1}{d_1 d'_1} + \frac{d'_2}{d_2 d'_2} + \dots + \frac{1}{v(\pi)} \right] \\ &= (N-c) \left[\frac{d'_1 + d'_2 + \dots + 1}{v(\pi)} \right] \\ &= \frac{N-c}{v(\pi)} [\sigma(v(\pi)) - v(\pi)] \end{aligned}$$

which is $\mathcal{O}(N)$ since $|\pi| = c = \mathcal{O}(1)$ (i.e. number of strands $c = \mathcal{O}(1)$). ◀

If S is a connected unpseudoknotted secondary structure with ordering π , you can go from any base i to j in two different paths around the circumference of $\text{Poly}(S, \pi)$ (clockwise or anticlockwise). We define the *length* function $l[i, j]$ to be the length of the shorter path, including both i and j as follows:

$$l[i, j] = \min\{|i - j| + 1, N - |i - j| + 1\} \quad (4)$$

Also, $\llbracket i, j \rrbracket$ is used to denote that shorter *segment* of length $l[i, j]$, where the direction from base i to base j is the same as the system strands' direction.

3.2 How to slice a pizza (secondary structure)

We want to slice any R -fold rotationally symmetric secondary structure S , **like pizza**, to the centre of its $\text{Poly}(S, \pi)$, without intersecting any of its base pairs. First, we formalise (Definition 15) a special type of backbone cut, called an *admissible R -symmetric backbone cut*. Then, we will prove (Lemma 16) its existence for S .

► **Definition 15** (Admissible R -symmetric backbone cut). *For any connected unpseudoknotted secondary structure S with strand ordering π , the R -symmetric backbone cut \mathcal{C}_R^b generated by b is admissible, if for all covalent bonds $x \in \mathcal{C}_R^b$, x is not “enclosed” by any base pair $(i, j) \in \text{Poly}(S, \pi)$, more formally: $x \not\subseteq \llbracket i, j \rrbracket$. An example is shown in Figure 3.*

► **Lemma 16.** *For any R -fold rotationally symmetric secondary structure S , there exists at least one admissible R -symmetric backbone cut of S .*

Proof. From $\text{Poly}(S, \pi)$, select a base pair (i, j) that has maximal length $l[i, j]$, at least one of $[i - 1, i]$ and $[j, j + 1]$ must be a covalent bond, otherwise if both were nicks, S would be disconnected (a contradiction). We claim that this covalent bond, which we denote $[a, a + 1]$, is an admissible R -symmetric backbone cut generator, otherwise there exists a base pair $(m, n) \in \text{Poly}(S, \pi)$ such that $[a, a + 1] \subseteq \llbracket m, n \rrbracket$, giving a contradiction by either: (a) $\llbracket m, n \rrbracket$ must contain $\llbracket i, j \rrbracket$ which contradicts the maximality of $l(i, j)$, or (b) (m, n) and (i, j) intersect forming a pseudoknot. All covalent bonds in $\mathcal{C}_R^{[a, a+1]}$ have the same situation because of R -fold symmetry of S , which implies that $\mathcal{C}_R^{[a, a+1]}$ is an admissible R -symmetric backbone cut of S . ◀

► **Lemma 17.** *For any connected unpseudoknotted secondary structure S , if there exists at least one base pair (i, j) such that $\llbracket i, j \rrbracket > \frac{N}{R}$, then S can not be R -fold rotationally symmetric.*

Note that any R -fold rotationally symmetric secondary structure S can have more than one admissible R -symmetric cut. Before defining what we mean by a pizza slice (formally, symmetric slice in Definition 19), the following lemma ensures such a slice is connected.

► **Lemma 18 (Pizza slicing lemma).** *For any $R \geq 2$ and any R -fold rotationally symmetric secondary structure S , let G be the graph obtained from $\text{Poly}(S, \pi)$ by removing the covalent bonds of admissible R -symmetric backbone cut \mathcal{C}_R^b generated by any covalent bond b , such that $G = (V(\text{Poly}(S, \pi)), E(\text{Poly}(S, \pi)) \setminus \mathcal{C}_R^b)$, then G is disconnected and consists exactly of R connected isomorphic components.*

Proof. Lemma 16 ensures the existence of at least one admissible R -symmetric backbone cut of S , assume it is generated by $b = [i_{A_m^n}, (i + 1)_{A_m^n}]$, then by Definition 12: $\mathcal{C}_R^b = \{[i_{A_m^{a(n)}}, (i + 1)_{A_m^{a(n)}}] : \forall a \in H \leq G^\pi, |H| = R\}$. For any two (recall, $R \geq 2$) “consecutive” covalent bonds in \mathcal{C}_R^b (formally: $b_k = [i_{A_m^{a_k(n)}}, (i + 1)_{A_m^{a_k(n)}}]$ and $b_{k+1} = [i_{A_m^{a_{k+1}(n)}}, (i + 1)_{A_m^{a_{k+1}(n)}}]$), we construct the “pizza slice” G_k to be the subgraph of $\text{Poly}(S, \pi)$ induced by all vertices (or bases) that belong to $I_k = \llbracket (i + 1)_{A_m^{a_k(n)}}, i_{A_m^{a_{k+1}(n)}} \rrbracket$. Intuitively, G_k is the slice we get after cutting $\text{Poly}(S, \pi)$ at b_k and b_{k+1} . At the end we will have a sequence of subgraphs $\mathcal{G} = \{G_1, G_2, \dots, G_R\}$. Because of symmetry, all subgraphs in \mathcal{G} are isomorphic. We claim that each subgraph in \mathcal{G} is exactly one connected component, and $G = \bigcup_{k=1}^R G_k$.

For any G_k , first we will show that G_k is disconnected from any other G_l with $l \neq k$, which follows from two observations: From Lemma 17, we know that the length of $I_k < \frac{N}{R}$ and I_1, \dots, I_R are disjoint segments by construction (via symmetry R), hence G_k and G_l are not connected by a covalent bond. To see that there is no base pair (x, y) connecting G_k and G_l : assume for the sake of contradiction that such a base pair (x, y) exists in $\text{Poly}(S, \pi)$, then one of the two covalent bonds b_k or $b_{k+1} \subseteq \llbracket x, y \rrbracket$ (by the definition of b_k, b_{k+1} above), contradicting the fact that \mathcal{C}_R^b is an admissible backbone cut. Also, it follows directly that $G = \bigcup_{k=1}^R G_k$ from the definition of inducing subgraphs.

We next wish to show that each slice G_k is connected. First, there exists $d \geq 1$ such that G consists of dR connected components, because for each k , $G_k \in \mathcal{G}$ is isomorphic (so if G_k has $d \geq 1$ components then all $G_l \in \mathcal{G}$ do also). Next we will show $d = 1$. Since G is the graph obtained from the connected graph $\text{Poly}(S, \pi)$ by removing $|\mathcal{C}_R^b| = R$ covalent bonds, so there are only two cases: (a) if each of these R covalent bonds is a cut edge, or bridge [40], G will consist of $R + 1$ components, contradicting the fact that number of its components must be dR for some $d \geq 1$, so $d = 1$, implying that each subgraph in \mathcal{G} consists of exactly one component. (b) The only other case is that G has R components (since we’ve already shown that the R slices are not connected to each other), giving the lemma statement. ◀

► **Definition 19** (Symmetric slice). *From the construction in Lemma 18, each of the R isomorphic subgraphs (components) in \mathcal{G} is called a symmetric slice of $\text{Poly}(S, \pi)$, denoted by \triangleright^S . Also, the loop free energy of a symmetric slice is:*

$$\Delta G(\triangleright^S) = \sum_{l \in \triangleright^S} \Delta G(l) \quad (5)$$

For any R -fold symmetric secondary structure S , the following lemma shows the existence of a unique loop in the center of $\text{Poly}(S, \pi)$ surrounded by the outer base pairs of its symmetric slices. We call it the *central loop* of S , and denote it by \bigcirc^S . This central loop plays a crucial role in validating our slicing and swapping strategy (Lemmas 22 and 23) and determining the exact upper bound (Lemma 25) of symmetric secondary structures need to be backtracked.

► **Lemma 20.** *For any $(R \geq 2)$ -fold symmetric secondary structure S , there exists a single loop, that we call the central loop \bigcirc^S , that is not contained in any of the R symmetric slices. If $R > 2$ then \bigcirc^S is a multiloop, if $R = 2$ then \bigcirc^S is either a multiloop, stack, or an internal loop.*

Proof. Using our construction in Lemma 18, let $G_k \in \mathcal{G}$ be any symmetric slice, and assume a local indexing from 1 to $\frac{N}{R}$ for the bases in the segment I_k . Let $\mathcal{N}_k = \{(m, n) \in G_k : \neg \exists (m', n') \in G_k \text{ such that } m' < m < n < n'\}$. Intuitively, \mathcal{N}_k contains all outer base pairs of G_k , we call \mathcal{N}_k the *nesting set* of G_k as it determines how G_k will geometrically look like (when staring at it from the centre of the pizza).

Intuitively, we construct a path P_k as the concatenation of a segment of covalent bonds from G_k , and then a base pair in \mathcal{N}_k , then more covalent bonds, then a base pair, and so on. Formally, let $d = |\mathcal{N}_k|$, then $P_k = \llbracket 1, m_1 \rrbracket . (m_1, n_1) . \llbracket n_1, m_2 \rrbracket . (m_2, n_2) \dots (m_c, n_c) . \llbracket n_c, \frac{N}{R} \rrbracket$. Note that $\llbracket 1, m_1 \rrbracket$ and $\llbracket n_c, \frac{N}{R} \rrbracket$ may be substrands of length zero. Using this construction, P_k must be a path, otherwise G_k will be a disconnected graph contradicting Lemma 18.

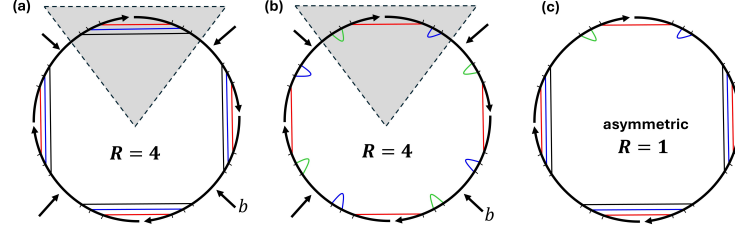
Now, we will construct this central loop \bigcirc^S as follows: for simplicity of notation, let $\mathcal{C}_R^b = \{b, b_2, \dots, b_R\}$, then $\bigcirc^S = b.P_1.b_2.P_2 \dots b_R.P_R$. If $R > 2$, \bigcirc^S must be a multiloop, since a multiloop is defined by being bordered by > 2 base pairs (hydrogen bonds). If $R = 2$, \bigcirc^S will be a multiloop if and only if $|\mathcal{N}_k| \geq 2$ (this implies there are $R|\mathcal{N}_k| = 2|\mathcal{N}_k| > 2$ base pairs bordering the central loop), otherwise \bigcirc^S will be an internal loop or stack. \bigcirc^S can not be external loop otherwise this will contradict the connectedness of S , nor can \bigcirc^S be a bulge nor hairpin loop as either of these will contradict the symmetry of S . ◀

Because of the crucial importance of multiloops for our strategy, we highlight the multiloop energy model that is used in the standard dynamic programming algorithms. The free energy of a multiloop has the following linear form [10]:

$$\Delta G^{\text{multi}} = \Delta G_{\text{init}}^{\text{multi}} + b\Delta G_{\text{bp}}^{\text{multi}} + n\Delta G_{\text{nt}}^{\text{multi}}, \quad (6)$$

where, $\Delta G_{\text{init}}^{\text{multi}}$ is called the penalty for formation of the multiloop, $\Delta G_{\text{bp}}^{\text{multi}}$ is called the penalty for each of its b base pairs that border the interior of the multiloop, and $\Delta G_{\text{nt}}^{\text{multi}}$ is called the penalty for each of the n free bases inside the multiloop. For any R -fold symmetric secondary structure S , $b\Delta G_{\text{bp}}^{\text{multi}}$ and $n\Delta G_{\text{nt}}^{\text{multi}}$ are shared equally between the R symmetric slices of $\text{Poly}(S, \pi)$, hence R divides both b and n . So, we denote $\Delta G^{\text{multi}} = \Delta G_{\text{init}}^{\text{multi}} + R(\Delta G_{\triangleright^S}^{\text{multi}})$, where $\Delta G_{\triangleright^S}^{\text{multi}}$ is the energy contribution of each symmetric slice of $\text{Poly}(S, \pi)$ to the multiloop free energy, such that $\Delta G_{\triangleright^S}^{\text{multi}} = \frac{b}{R}\Delta G_{\text{bp}}^{\text{multi}} + \frac{n}{R}\Delta G_{\text{nt}}^{\text{multi}}$.

► **Note 21.** For any connected unpseudoknotted secondary structure S of c strands, we use $\overline{\Delta G}(S)$ to denote the snMFE of S , in other words $\overline{\Delta G}(S) = \sum_{l \in S} \Delta G(l) + (c - 1)\Delta G^{\text{assoc}}$. It is clear that $\overline{\Delta G}(S) \leq \Delta G(S)$, as the symmetry correction $k_B T \log R \geq 0$.



■ **Figure 3** Slicing and swapping strategy for constructing new asymmetric structure by combining two symmetric structures with the same symmetric backbone cut. (a) 4-fold symmetric secondary structure S_i , with admissible 4-symmetric backbone cut C_R^b . Black arrows: indicate the four covalent bonds forming C_R^b generated by the covalent bond b . (b) 4-fold symmetric secondary structure S_j , sharing the same cut C_R^b as S_i . Black arrows: indicate the four covalent bonds forming C_R^b . (c) Asymmetric secondary structure S_k that is constructed by replacing the grey shaded “slice” from S_i by its corresponding slice from S_j , using the proof of Lemma 22.

Intuitively, the following lemma lets us take two symmetric pizzas with the same admissible symmetric cut for which we know their snMFE, and swap a slice from one into the other to get a new asymmetric pizza whose true MFE lies between their snMFEs. The key intuition is that we are transforming two symmetric secondary structures into an asymmetric one.

► **Lemma 22** (Free-energy sandwich theorem for two R -fold rotationally symmetric structures). *For any two distinct ($R \geq 3$)-fold rotationally symmetric secondary structures, S_i and S_j , of c strands, such that $R \geq 3$ and $\overline{\Delta G}(S_i) \leq \overline{\Delta G}(S_j)$ and S_i and S_j have the same R -admissible backbone cut C_R^b , then there exists at least one asymmetric secondary structure S_k , such that $\overline{\Delta G}(S_i) \leq \Delta G(S_k) \leq \overline{\Delta G}(S_j)$. Furthermore, the statement holds if $R = 2$ and both the central loops $\bigcirc^{S_i}, \bigcirc^{S_j}$ are multiloops.*

Proof. If $R \geq 3$, from Lemma 20, there exist two unique central multiloops for S_i and S_j , denoted by \bigcirc^{S_i} and \bigcirc^{S_j} . Also, if $R = 2$, by hypothesis, \bigcirc^{S_i} and \bigcirc^{S_j} are multiloops. We prove the claim with two cases:

1. $\overline{\Delta G}(S_i) = \overline{\Delta G}(S_j)$:

$$\sum_{l \in S_i} \Delta G(l) + (c-1)\Delta G^{\text{assoc}} = \sum_{l \in S_j} \Delta G(l) + (c-1)\Delta G^{\text{assoc}} \quad (7)$$

$$\sum_{l \in S_i} \Delta G(l) = \sum_{l \in S_j} \Delta G(l) \quad (8)$$

$$R(\Delta G(\triangleright^{S_i})) + \Delta G(\bigcirc^{S_i}) = R(\Delta G(\triangleright^{S_j})) + \Delta G(\bigcirc^{S_j}) \quad (9)$$

$$R(\Delta G(\triangleright^{S_i})) + R\Delta G_{\triangleright^{S_i}}^{\text{multi}} + \Delta G_{\text{init}}^{\text{multi}} = R(\Delta G(\triangleright^{S_j})) + R\Delta G_{\triangleright^{S_j}}^{\text{multi}} + \Delta G_{\text{init}}^{\text{multi}} \quad (10)$$

$$R(\Delta G(\triangleright^{S_i})) + \Delta G_{\triangleright^{S_i}}^{\text{multi}} = R(\Delta G(\triangleright^{S_j})) + \Delta G_{\triangleright^{S_j}}^{\text{multi}} \quad (11)$$

$$\Delta G(\triangleright^{S_i}) + \Delta G_{\triangleright^{S_i}}^{\text{multi}} = \Delta G(\triangleright^{S_j}) + \Delta G_{\triangleright^{S_j}}^{\text{multi}} \quad (12)$$

We define a new secondary structure S_k using our slicing and swapping strategy, shown in Figure 3, by removing one slice from S_i , and adding its corresponding slice from S_j . Lemma 18 guarantees that the new secondary structure S_k is connected and unpsudoknotted. Furthermore, S_k is asymmetric since $S_i \neq S_j$. S_k being asymmetric means that its

rotational symmetry $R_k = 1$, $k_B T \log R_k = 0$ hence we can write:

$$\begin{aligned}\Delta G(S_k) &= (R - 1) (\Delta G(\triangleright^{S_i}) + \Delta G_{\triangleright^{S_i}}^{\text{multi}}) + G_{\text{init}}^{\text{multi}} + (c - 1) \Delta G^{\text{assoc}} \\ &\quad + (\Delta G(\triangleright^{S_j}) + \Delta G_{\triangleright^{S_j}}^{\text{multi}}) \\ \Delta G(S_k) &= R (\Delta G(\triangleright^{S_i}) + \Delta G_{\triangleright^{S_i}}^{\text{multi}}) + \Delta G_{\text{init}}^{\text{multi}} + (c - 1) \Delta G^{\text{assoc}}\end{aligned}$$

with the final step using Equation (12). Hence $\overline{\Delta G}(S_i) = \Delta G(S_k) = \overline{\Delta G}(S_j)$.

2. $\overline{\Delta G}(S_i) < \overline{\Delta G}(S_j)$: Following the same algebraic manipulation as Equation (7) to Equation (12), but replacing $=$ with $<$, we get the following:

$$\Delta G(\triangleright^{S_i}) + \Delta G_{\triangleright^{S_i}}^{\text{multi}} < \Delta G(\triangleright^{S_j}) + \Delta G_{\triangleright^{S_j}}^{\text{multi}}$$

As before, we define a new connected asymmetric secondary structure S_k , using the same slicing and swapping strategy, resulting in: $\overline{\Delta G}(S_i) < \Delta G(S_k) < \overline{\Delta G}(S_j)$. ◀

Lemma 22 states that, if two symmetric secondary structures, having the same admissible R-symmetric backbone cut, belong to the same energy level based on the symmetry-naive MFE algorithm, ignoring symmetry entropic correction, this implies the existence of at least one asymmetric secondary structure that actually belong to the same energy level because symmetry correction for asymmetric structures is zero. If the two secondary structures belong to two different energy levels, then there exist at least one asymmetric secondary structure that actually belongs to an energy level that strictly lies between the other two energy levels.

Intuition for the case of $R = 2$ and the central loop is not a multiloop

When $R = 2$, and the central loop is not a multiloop, the proof of Lemma 22 breaks. From Lemma 20, when $R = 2$ the central loop is either a multiloop, internal loop or stack loop. The multiloop case has been handled already (Lemma 22), and the stack loop case can be subsumed into the internal loop case, since stacks are considered a special type of internal loop in the standard energy model [10]. Instead of depending on having the same admissible 2-symmetric backbone cut, we depend on sharing the same central internal loop itself, this more restricted hypothesis implies having the same admissible 2-symmetric backbone cut too, allowing us to prove Lemma 23 using a similar strategy to Lemma 22.

► **Lemma 23** (Free-energy sandwich theorem for two 2-fold rotationally symmetric structures). *For any two distinct 2-fold rotationally symmetric secondary structures, S_i and S_j , of c strands, such that $\overline{\Delta G}(S_i) \leq \overline{\Delta G}(S_j)$ and both have the same central internal loop $\bigcirc^{S_i} = \bigcirc^{S_j}$, then there exists at least one asymmetric secondary structure S_k , such that $\overline{\Delta G}(S_i) \leq \Delta G(S_k) \leq \overline{\Delta G}(S_j)$.*

Proof. Since $\bigcirc^{S_i} = \bigcirc^{S_j}$, then both have the same admissible 2-symmetric backbone cut as any covalent bond b that belong to any side of the internal loop can be its generator.

$$\begin{aligned}\overline{\Delta G}(S_i) &\leq \overline{\Delta G}(S_j) \\ \sum_{l \in S_i} \Delta G(l) + (c - 1) \Delta G^{\text{assoc}} &\leq \sum_{l \in S_j} \Delta G(l) + (c - 1) \Delta G^{\text{assoc}} \\ \sum_{l \in S_i} \Delta G(l) &\leq \sum_{l \in S_j} \Delta G(l) \\ 2(\Delta G(\triangleright^{S_i})) + \Delta G(\bigcirc^{S_i}) &\leq 2(\Delta G(\triangleright^{S_j})) + \Delta G(\bigcirc^{S_j}) \\ \Delta G(\triangleright^{S_i}) &\leq \Delta G(\triangleright^{S_j})\end{aligned}$$

We define a new secondary structure S_k using our slicing and swapping strategy, shown in Figure 3, by removing one slice (half in this case) from S_i , and adding its corresponding slice from S_j . Note that S_k will have also the same central internal loop $\bigcirc^{S_k} = \bigcirc^{S_i}$. Lemma 18 guarantees that S_k is connected and unpseudoknotted. Since S_k is asymmetric:

$$\Delta G(S_k) = \Delta G(\triangleright^{S_i}) + \Delta G(\triangleright^{S_j}) + \Delta G(\bigcirc^{S_k}) + (c-1)\Delta G^{\text{assoc}}$$

Which implies that $\overline{\Delta G}(S_i) \leq \Delta G(S_k) \leq \overline{\Delta G}(S_j)$. \blacktriangleleft

We now have two sandwich theorems that we can use to construct an asymmetric structure: Lemmas 22 and 23. In Section 4 we give a backtracking algorithm to search for suitable S_i and S_j , with the goal of applying either one of these two sandwich theorems to S_i and S_j . To get an overall polynomial bound for the backtracking algorithm, we wish to bound, given S_i , how many secondary structures to scan before finding a suitable S_j . Lemma 14 gives an upper bound on this number when applying Lemma 22. Next, Lemma 24 gives this upper bound when applying Lemma 23.

Unfortunately, the bound in Lemma 24 is larger than Lemma 14, since the energy model is more complex for internal loops than multiloops [11].

► **Lemma 24** (Upper bound on number of central internal loops). *For any set of c strands with specific ordering π , for any set \mathcal{T} of 2-fold rotationally symmetric secondary structures ($R = 2$), such that each has a distinct internal central loop, the cardinality of \mathcal{T} , $|\mathcal{T}| \leq \sum_{s \in y} (\|A\|_s \|T\|_s + \|G\|_s \|C\|_s) \leq N^2/16$, where y is a fundamental component of π , such that $\pi = y^2$, and $\|B\|_s$ denotes the number of bases in strand s of type B for all $B \in \{A, T, G, C\}$.*

Proof. Let \mathcal{T} be any set of 2-fold rotationally symmetric secondary structures. Since each $S \in \mathcal{T}$ has a distinct internal central loop, we focus only on giving an upper bound on the maximum number of distinct internal central loops. Since $R = 2$, the two base pairs forming any central internal loop must be within two strands of the same type X of the same order m within their fundamental components (for example the strands are X_m^1 and X_m^2), otherwise we have a disconnected secondary structure due to the existence of nicks.

Only one of the two base pairs of any internal central loop needs to be specified explicitly, since, by symmetry, the other base pair is automatically determined. So, by considering all base pairs (including many that are irrelevant), the number of all distinct central internal loops is $\leq \sum_{s \in y} (\|A\|_s \|T\|_s + \|G\|_s \|C\|_s)$. Hence, $|\mathcal{T}| \leq \sum_{s \in y} (\|A\|_s \|T\|_s + \|G\|_s \|C\|_s)$. Using the two following number theoretic facts:

- If we have two indistinguishable strands, of the same type, X , of length n , the maximum intra-base pairs between them happens when the sequence of X is a word over $\{A, T\}$ or $\{G, C\}$ such that $\|A\|_X = \lfloor \frac{n}{2} \rfloor$ and $\|T\|_X = \lceil \frac{n}{2} \rceil$ or vice versa, and the same for $\{G, T\}$.
- For any integer $n > 0$, if $n = n_1 + n_2 + \dots + n_k$, such that $n_i \geq 0$ for all $i \in \{1, 2, \dots, k\}$, then $n^2 \geq n_1^2 + n_2^2 + \dots + n_k^2$.

We get the following:

$$\begin{aligned} \sum_{s \in y} (\|A\|_s \|T\|_s + \|G\|_s \|C\|_s) &\leq \left\lceil \frac{|s_1|}{2} \right\rceil \left\lfloor \frac{|s_1|}{2} \right\rfloor + \left\lceil \frac{|s_2|}{2} \right\rceil \left\lfloor \frac{|s_2|}{2} \right\rfloor + \dots + \left\lceil \frac{|s_{c/2}|}{2} \right\rceil \left\lfloor \frac{|s_{c/2}|}{2} \right\rfloor \\ &\leq \left(\frac{|s_1|}{2} \right)^2 + \left(\frac{|s_2|}{2} \right)^2 + \dots + \left(\frac{|s_{c/2}|}{2} \right)^2 \\ &= \frac{|s_1|^2 + |s_2|^2 + \dots + |s_{c/2}|^2}{4} \\ &\leq \frac{(N/2)^2}{4} = \frac{N^2}{16} \end{aligned}$$

where $s_i, i \in \{1, \dots, c\}$, are strand types. Hence: $|\mathcal{T}| \leq \sum_{s \in y} (\|A\|_s \|T\|_s + \|G\|_s \|C\|_s) \leq N^2/16$. \blacktriangleleft

3.3 Polynomial upper bound on number of symmetric secondary structures (for future backtracking)

► **Lemma 25.** *Given an ordering π of c strands, for any set \mathcal{T} of distinct symmetric secondary structures such that*

1. *for any two $(R > 2)$ -fold symmetric secondary structures $S_i, S_j \in \mathcal{T}$, where S_i and S_j have different admissible R -symmetric backbone cuts (we mean all possible cuts are different), and*
2. *for any two 2-fold symmetric secondary structures $S_i, S_j \in \mathcal{T}$, where S_i and S_j have different admissible R -symmetric backbone cuts (all possible cuts are different) or different central internal loops,*

then $|\mathcal{T}| \leq \mathcal{U}$, where $\mathcal{U} = \frac{N-c}{v(\pi)} [\sigma(v(\pi)) - v(\pi)] + \frac{N^2}{16} = \mathcal{O}(N^2)$.

Proof. The proof is a trivial implication of Lemmas 14 and 24. (More formally: Let $\mathcal{U} = \mathcal{U}_1 + \mathcal{U}_2$ where \mathcal{U}_1 is the upper bound on the number of on unique symmetric backbone cuts (Lemma 14), and \mathcal{U}_2 is the upper bound on the number of unique central internal loops (Lemma 24). Assume for the sake of contradiction that $|\mathcal{T}| > \mathcal{U}$. From the pigeon hole principle, $|\mathcal{T}| > \mathcal{U}$ implies repeating at least one symmetric backbone cut or a central internal loop contradicting the hypothesis about structures of \mathcal{T} .) ◀

The (bad) quadratic bound in Lemma 24 is not that frequent: In particular, that bound only appears when $R = 2$ and the central loop is an internal loop for both symmetric secondary structures (since $R = 2$ this implies that the repetition number for every strand type is *even*, which in practice, say, for random or typical systems, may not be frequent). In particular the following lemma gives a *linear* bound when the repetition number of at least one strand type is odd.

► **Lemma 26.** *For any R -fold rotationally symmetric secondary structure S , with ordering π , such that R is even, then the repetition number of each strand type must be even. Hence for any system of c strands (k strand types) such that the repetition number of some strand type is odd, then \mathcal{U} , where $\mathcal{U} = \frac{N-c}{v(\pi)} [\sigma(v(\pi)) - v(\pi)] = \mathcal{O}(N)$.*

Proof. Suppose S is a R -fold rotationally symmetric such that R is even. R divides $v(\pi)$ (see Lemma 13), which implies $v(\pi)$ is even too. Since $\pi = x^{v(\pi)}$ then $\pi = y^2$ such that $y = x^{v(\pi)/2}$, and this is valid only if the repetition number of each strand type is even. The linearity of \mathcal{U} follows directly if repetition number of at least one strand type is odd. ◀

4 Backtracking to find the true MFE

The full version of this work [33] contains the technical details of the backtracking procedure used to prove Theorem 1 below, specifically see Section 4, and Algorithm 2 of Appendix C [33]. The complexity of our backtracking algorithm, is summarized in the following lemma.

► **Lemma 27.** *The running time of the backtracking algorithm (Algorithm 2 in Appendix C in the full version of this paper [33]), for a set of $c = \mathcal{O}(1)$ DNA or RNA strands of total length N bases, is $\mathcal{O}(N^4(c-1)!)$, and it uses $\mathcal{O}(N^4)$ space.*

5 Time and space analysis of MFE algorithm

► **Theorem 1.** *There is an $\mathcal{O}(N^4(c-1)!)$ time and $\mathcal{O}(N^4)$ space algorithm for the Minimum Free Energy unpseudoknotted secondary structure prediction problem, including rotational symmetry, for a set of $c = \mathcal{O}(1)$ DNA or RNA strands of total length N bases.*

Proof. The snMFE algorithm of Dirks et al. runs in time $\mathcal{O}(N^3(c-1)!)$ and space $\mathcal{O}(N^2)$ [10]. In Algorithm 1 in Appendix B of the full version [33], we give their snMFE pseudocode but augmented with three matrices $M^{\text{b:int}}$, $M^{\text{b:mul}}$, and $M^{\text{m:2}}$, with no asymptotic change to run time but an increase to $\mathcal{O}(N^3)$ space. Also, by Lemma 27, the time complexity of our backtracking algorithm, Algorithm 2, is $\mathcal{O}(N^4(c-1)!)$, and the space complexity is $\mathcal{O}(N^4)$.

Hence after running both algorithms we get an $\mathcal{O}(N^4(c-1)!)$ algorithm for the MFE unpsudoknotted secondary structure prediction problem, including rotational symmetry, with space complexity $\mathcal{O}(N^4)$. ◀

By Remark 32 in Section 4 in the full version [33], there is a time-space trade-off yielding an alternative algorithm that runs in $\mathcal{O}((N^4 \log N)(c-1)!)$ time and $\mathcal{O}(N^3)$ space.

References

- 1 Tatsuya Akutsu. Dynamic programming algorithms for RNA secondary structure prediction with pseudoknots. *Discrete Applied Mathematics*, 104(1-3):45–62, 2000. doi:10.1016/S0166-218X(00)00186-4.
- 2 Peter William Atkins, Julio De Paula, and James Keeler. *Atkins' physical chemistry*. Oxford university press, 2023.
- 3 Kimon Boehmer, Sarah J Berkemer, Sebastian Will, and Yann Ponty. RNA triplet repeats: Improved algorithms for structure prediction and interactions. *24th International Workshop on Algorithms in Bioinformatics (WABI 2024)*, 2024. URL: <https://hal.science/hal-04589903>.
- 4 Edward Bormashenko. Entropy, information, and symmetry: Ordered is symmetrical. *Entropy*, 22(1):11, 2019. doi:10.3390/E22010011.
- 5 Richard A Brualdi. *Introductory combinatorics*. Pearson Education India, 1977.
- 6 Gourab Chatterjee, Neil Dalchau, Richard A. Muscat, Andrew Phillips, and Georg Seelig. A spatially localized architecture for fast and modular DNA computing. *Nature Nanotechnology*, 12(9):920–927, September 2017. doi:10.1038/nnano.2017.127.
- 7 Ho-Lin Chen, Anne Condon, and Hosna Jabbari. An $\mathcal{O}(n^5)$ algorithm for MFE prediction of kissing hairpins and 4-chains in nucleic acids. *Journal of Computational Biology*, 16(6):803–815, 2009. doi:10.1089/CMB.2008.0219.
- 8 Alexander Churkin, Matan Drory Retwitzer, Vladimir Reinharz, Yann Ponty, Jérôme Waldispühl, and Danny Barash. Design of RNAs: comparing programs for inverse RNA folding. *Briefings in bioinformatics*, 19(2):350–358, 2018. doi:10.1093/BIB/BBW120.
- 9 Anne Condon, Monir Hajiaghayi, and Chris Thachuk. Predicting minimum free energy structures of multi-stranded nucleic acid complexes is APX-hard. In *27th International Conference on DNA Computing and Molecular Programming (DNA 27)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.
- 10 Robert M Dirks, Justin S Bois, Joseph M Schaeffer, Erik Winfree, and Niles A Pierce. Thermodynamic analysis of interacting nucleic acid strands. *SIAM review*, 49(1):65–88, 2007. doi:10.1137/060651100.
- 11 Robert M Dirks and Niles A Pierce. A partition function algorithm for nucleic acid secondary structure including pseudoknots. *Journal of computational chemistry*, 24(13):1664–1677, 2003. doi:10.1002/JCC.10296.
- 12 Robert M Dirks and Niles A Pierce. An algorithm for computing nucleic acid base-pairing probabilities including pseudoknots. *Journal of computational chemistry*, 25(10):1295–1304, 2004. doi:10.1002/JCC.20057.
- 13 David Doty and Benjamin Lee. nuad: Nucleic acid designer, March 2022. Uses, and generalises, sequence design principles from [42]. URL: <https://github.com/UC-Davis-molecular-computing/nuad>.

- 14 Joshua Fern and Rebecca Schulman. Design and characterization of dna strand-displacement circuits in serum-supplemented cell medium. *ACS Synthetic Biology*, 6(9):1774–1783, 2017. doi:10.1021/acssynbio.7b00105.
- 15 Mark E Fornace, Nicholas J Porubsky, and Niles A Pierce. A unified dynamic programming framework for the analysis of interacting nucleic acid strands: enhanced models, scalability, and speed. *ACS Synthetic Biology*, 9(10):2665–2678, 2020.
- 16 Cody Geary, Paul WK Rothmund, and Ebbe S Andersen. A single-stranded architecture for cotranscriptional folding of RNA nanostructures. *Science*, 345(6198):799–804, 2014.
- 17 Ivo L Hofacker, Christian M Reidys, and Peter F Stadler. Symmetric circular matchings and RNA folding. *Discrete mathematics*, 312(1):100–112, 2012. doi:10.1016/J.DISC.2011.06.004.
- 18 Hosna Jabbari, Ian Wark, Carlo Montemagno, and Sebastian Will. Knotty: efficient and accurate prediction of complex RNA pseudoknot structures. *Bioinformatics*, 34(22):3849–3856, 2018. doi:10.1093/BIOINFORMATICS/BTY420.
- 19 Ronny Lorenz, Stephan H Bernhart, Christian Höner zu Siederdissen, Hakim Tafer, Christoph Flamm, Peter F Stadler, and Ivo L Hofacker. ViennaRNA package 2.0. *Algorithms for molecular biology*, 6:1–14, 2011.
- 20 Rune B Lyngsø and Christian NS Pedersen. Pseudoknots in RNA secondary structures. In *Proceedings of the fourth annual international conference on Computational molecular biology*, pages 201–209, 2000.
- 21 Rune B Lyngsø and Christian NS Pedersen. RNA pseudoknot prediction in energy-based models. *Journal of computational biology*, 7(3-4):409–427, 2000. doi:10.1089/106652700750050862.
- 22 David H Mathews, Jeffrey Sabina, Michael Zuker, and Douglas H Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *Journal of molecular biology*, 288(5):911–940, 1999.
- 23 John S McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers: Original Research on Biomolecules*, 29(6-7):1105–1119, 1990.
- 24 Ruth Nussinov and Ann B Jacobson. Fast algorithm for predicting the secondary structure of single-stranded RNA. *Proceedings of the National Academy of Sciences*, 77(11):6309–6313, 1980.
- 25 Ruth Nussinov, George Pieczenik, Jerrold R Griggs, and Daniel J Kleitman. Algorithms for loop matchings. *SIAM Journal on Applied mathematics*, 35(1):68–82, 1978.
- 26 Lulu Qian and Erik Winfree. Scaling up digital circuit computation with DNA strand displacement cascades. *Science*, 332(6034):1196–1201, 2011.
- 27 Jens Reeder and Robert Giegerich. Design, implementation and evaluation of a practical pseudoknot folding algorithm based on thermodynamics. *BMC bioinformatics*, 5:1–12, 2004.
- 28 Elena Rivas and Sean R Eddy. A dynamic programming algorithm for RNA structure prediction including pseudoknots. *Journal of molecular biology*, 285(5):2053–2068, 1999.
- 29 John SantaLucia Jr. A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. *Proceedings of the National Academy of Sciences*, 95(4):1460–1465, 1998.
- 30 John SantaLucia Jr and Donald Hicks. The thermodynamics of DNA structural motifs. *Annu. Rev. Biophys. Biomol. Struct.*, 33:415–440, 2004.
- 31 Joe Sawada. A fast algorithm to generate necklaces with fixed content. *Theoretical Computer Science*, 301(1-3):477–489, 2003. doi:10.1016/S0304-3975(03)00049-5.
- 32 Georg Seelig, David Soloveichik, David Yu Zhang, and Erik Winfree. Enzyme-free nucleic acid logic circuits. *science*, 314(5805):1585–1588, 2006.
- 33 Ahmed Shalaby and Damien Woods. An efficient algorithm to compute the minimum free energy of interacting nucleic acid strands, 2024. arXiv preprint arXiv:2407.09676. doi:10.48550/arXiv.2407.09676.

- 34 Robert J Silbey, Robert A Alberty, George A Papadantonakis, and Mounqi G Bawendi. *Physical chemistry*. John Wiley & Sons, 2022.
- 35 Anupama J. Thubagere, Wei Li, Robert F. Johnson, Zibo Chen, Shayan Doroudi, Yae Lim Lee, Gregory Izatt, Sarah Wittman, Niranjan Srinivas, Damien Woods, Erik Winfree, and Lulu Qian. A cargo-sorting DNA robot. *Science*, 357(6356), 2017.
- 36 Ignacio Tinoco, Olke C Uhlenbeck, and Mark D Levine. Estimation of secondary structure in ribonucleic acids. *Nature*, 230(5293):362–367, 1971.
- 37 Yasuo Uemura, Aki Hasegawa, Satoshi Kobayashi, and Takashi Yokomori. Tree adjoining grammars for RNA structure prediction. *Theoretical computer science*, 210(2):277–303, 1999. doi:10.1016/S0304-3975(98)00090-5.
- 38 Boya Wang, Siyuan Stella Wang, Cameron Chalk, Andrew D Ellington, and David Soloveichik. Parallel molecular computation on digital data stored in DNA. *Proceedings of the National Academy of Sciences*, 120(37):e2217330120, 2023.
- 39 Michael S Waterman and Temple F Smith. Rapid dynamic programming algorithms for RNA secondary structure. *Advances in Applied Mathematics*, 7(4):455–464, 1986.
- 40 Douglas Brent West et al. *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River, 2001.
- 41 Sungwook Woo and Paul WK Rothmund. Programmable molecular recognition based on the geometry of DNA nanostructures. *Nature chemistry*, 3(8):620, 2011.
- 42 Damien Woods, David Doty, Cameron Myhrvold, Joy Hui, Felix Zhou, Peng Yin, and Erik Winfree. Diverse and robust molecular algorithms using reprogrammable DNA self-assembly. *Nature*, 567(7748):366–372, 2019. doi:10.1038/S41586-019-1014-9.
- 43 Joseph N Zadeh, Conrad D Steenberg, Justin S Bois, Brian R Wolfe, Marshall B Pierce, Asif R Khan, Robert M Dirks, and Niles A Pierce. Nupack: Analysis and design of nucleic acid systems. *Journal of computational chemistry*, 32(1):170–173, 2011. doi:10.1002/JCC.21596.
- 44 David Yu Zhang and Georg Seelig. Dynamic DNA nanotechnology using strand-displacement reactions. *Nature chemistry*, 3(2):103–113, 2011.
- 45 Michael Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic acids research*, 31(13):3406–3415, 2003. doi:10.1093/NAR/GKG595.
- 46 Michael Zuker and David Sankoff. RNA secondary structures and their prediction. *Bulletin of mathematical biology*, 46:591–621, 1984.
- 47 Michael Zuker and Patrick Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic acids research*, 9(1):133–148, 1981. doi:10.1093/NAR/9.1.133.