GEOMETRIC COMBINATORICS AND COMPUTATIONAL MOLECULAR BIOLOGY: BRANCHING POLYTOPES FOR RNA SEQUENCES

ELIZABETH DRELLICH, ANDREW GAINER-DEWAR, HEATHER A. HARRINGTON, QIJUN HE, CHRISTINE HEITSCH, AND SVETLANA POZNANOVIĆ

Abstract. Questions in computational molecular biology generate various discrete optimization problems, such as DNA sequence alignment and RNA secondary structure prediction. However, the optimal solutions are fundamentally dependent on the parameters used in the objective functions. The goal of a parametric analysis is to elucidate such dependencies, especially as they pertain to the accuracy and robustness of the optimal solutions. Techniques from geometric combinatorics, including polytopes and their normal fans, have been used previously to give parametric analyses of simple models for DNA sequence alignment and RNA branching configurations. Here, we present a new computational framework, and proof-of-principle results, which give the first complete parametric analysis of the branching polytopes for real RNA sequences.

1. Motivation

Over the past four decades, improvements in biotechnology have greatly accelerated the amount of biological sequence data available. Yet, a fundamental challenge in computational molecular biology remains to reliably infer functional information from the linear encoding of DNA, RNA, and protein molecules.

As articulated in the central dogma of molecular biology, genetic information is stored in DNA from which it is transcribed into messenger RNA, and then translated into proteins by ribosomal and transfer RNA. However, as always in biology, a wealth of complexity lurks below the surface of this basic principle. Historically, most interest was focused on DNA sequences (as the cellular genome) and protein structures (as the cellular machinery). Since the early 2000's, though, attention has increasingly turned to RNA as many more critical functions have been revealed, including gene splicing, editing, and regulation.

Like DNA, RNA is a sequence of nucleic acids, abbreviated `A`, `C`, `G`, and `U` (instead of `T`), which form the familiar Watson-Crick pairings. Unlike the canonical double-stranded DNA helix, most RNA molecules are naturally single-stranded and the intra-sequence base pairings are an integral component of the three-dimensional structure. This is in contrast to the more subtle amino acid interactions which

2010 *Mathematics Subject Classification.* 92D10.

Key words and phrases. parametric analysis, secondary structure, polytope.

The authors' collaboration was funded by the Mathematics Research Communities program of the AMS. H. A. Harrington gratefully acknowledges the support of EPSRC Fellowship EP/K041096/1. C. Heitsch was supported in part by a Career Award at the Scientific Interface from the Burroughs Wellcome Fund. S. Poznanović was also supported in part by NSF grant DMS-1312817.

govern the formation of protein structures. However, knowing the structure of a protein or RNA molecule is critical to understanding and manipulating its cellular functions.

The structure of an RNA molecule is understood hierarchically, from the linear biochemical chain through the planar set of canonical base pairings to the 3D molecule, whose structure includes more complicated tertiary interactions like pseudoknots and base triples. Given the prevalence of RNA sequence data and the difficulties, both experimental and computational, of determining 3D structures, the majority of attention has focused on the set of base pairs, known as the *secondary structure*. In particular, one seeks to answer: *What are the native base pairs for a given RNA sequence?* As explained below, it is possible to efficiently compute an optimal secondary structure under the current Nearest Neighbor Thermodynamic Model (NNTM). However, when these minimum free energy (MFE) predictions are compared to structures derived from information-theoretic means, the current gold-standard, the average accuracy for longer ribosomal RNA sequences is only 40%. Hence, it is critical to understand which aspects of RNA base pairing are not captured well by the NNTM.

One known weakness of the current model is the energy function which governs the branching of an RNA secondary structure. For computational reasons, the entropic cost is modeled as an affine function with three parameters. A very natural question to ask is: *How does the optimal secondary structure depend on the branching loop parameters?*

Such a question is an example of a parametric analysis, which illuminates the dependencies of the solution on the underlying optimization parameters. Moreover, as we illustrate, methods from geometric combinatorics, specifically polytopes and their normal fans, can be used to answer this question, and others including accuracy and stability.

2. Background

2.1. **Computational molecular biology.** Biological sequences can be modeled as strings over a finite alphabet. The questions related to DNA and RNA are amenable to analysis using discrete mathematical techniques from algebra, combinatorics, and topology. We refer the reader to the following references for mathematical approaches for studying computational molecular biology [2, 8, 12, 24, 25, 36]. Of particular interest here are techniques from geometric combinatorics, which can inform parametric analysis of models arising from optimization questions, such as DNA sequence alignment and RNA secondary structure prediction.

Previous parametric analyses in molecular biology have primarily focused on DNA. Given sequences of DNA, an important problem is to find regions that have been evolutionary conserved. The first step requires *aligning* the sequences by adding spaces to the sequences until they are the same length. The *DNA sequence alignment* problem is to find the best alignment between sequences given some scoring function [35]. The optimal solution can be determined via dynamic programming optimization which scores various features and provides global alignment. The optimization fundamentally depends on parameters, and the notion of 'best matching' sequences may depend on the value of these parameters. Techniques from geometric combinatorics can be used [5, 35] to analyze systematically how sequence alignment

depends on these parameters. The mathematical background will be presented in Section 2.2.

The problem we focus on here is prediction of the secondary structure of a single given RNA sequence. Unlike DNA, which typically forms a fully base-paired double helix, RNA typically exists as a single strand whose bases can pair with one another to form diverse and complex secondary structures. The secondary structure of a given RNA molecule serves as a scaffold for the 3D structure [32]. As shown in Fig. 1, an RNA secondary structure consists of stacked base pairs (*helices*) and single-stranded regions (*loops*). The number of base pairings in the loop, or the number of helices meeting the loop, define the *degree* of the loop and the type. *Hairpin loops* have degree 1, *internal loops* have degree 2, and *multibranch loops* have degree greater than 2; each RNA molecule also has a (possibly empty) *exterior loop*, which includes the unpaired bases not contained in other loops. Figure 1 shows a secondary structure with all of these substructures labeled.

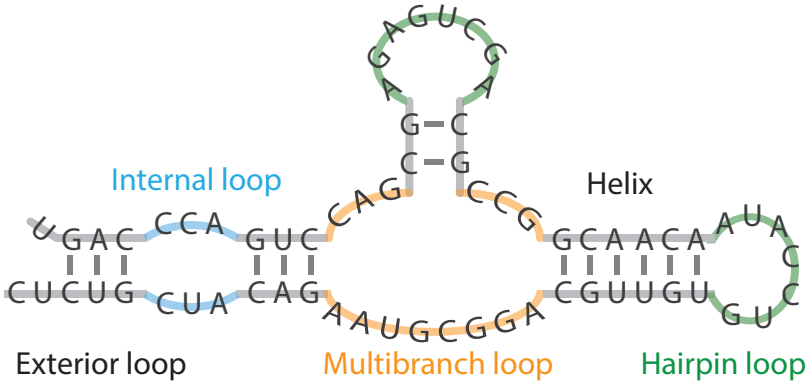


FIGURE 1. Secondary structure of RNA with substructures.

The base pairings in the secondary structure are difficult to determine experimentally, necessitating mathematical models to predict the structure [40]. Indeed secondary structure prediction has been an active area of research for several decades (see survey in [19]). One approach for predicting secondary structure is calculating the *free thermodynamic energy* of the RNA molecule according to the standard NNTM [17, 33]. In this model, bases that bond are generally energetically favorable and *stabilize* the RNA with a negative free energy, whereas unpaired bases are energetically disfavorable and *destabilize* the RNA with a positive free energy. Each substructure is assigned an energy value, and the total energy of the secondary structure is determined by summing over its substructures [17, 33].

If we further assume that secondary structure has no crossing base pairs, we could then generate secondary structure of an RNA sequence recursively over its subsequences. This allows us to compute the minimum energy structure efficiently using a dynamic programming algorithm. This approach is widely studied, and several software packages are available to perform the appropriate computations; widely-used implementations include `RNAfold` from the Vienna RNA package [13], `mfold` [41], and `RNAstructure` [26]. However, the optimal structure found using the thermodynamic model is often not the correct one. The prediction depends on the

thousands of parameter values in the objective function, affecting the substructure prediction.

Current versions of the scoring model [17,33] have over eight thousand parameters, but we focus our attention on three related to multibranch loops such as that at the center of Fig. 1. (We will discuss some details of the model and its parameters in Section 3.1.) We perform parametric analysis using geometric combinatorics, which has previously been applied to a simplified tree model of RNA base pairing that focuses on branching configurations [14], and extend this to study the branching of real RNA sequences under the Turner99 NNTM parameters. In the next section we introduce the required combinatorial and geometric notions to carry out this parametric analysis.

2.2. **Polytopes and geometric combinatorics.** The optimization of the NNTM objective function, with an emphasis on branching loop calculations, can be presented as a simple linear programming (LP) problem, and thus well-established mathematical tools can be used to tackle the optimization problem parametrically. To analyze optimization problems with large sets of feasible solutions, such as all possible secondary structures that could form from a particular sequence of RNA, a (relatively) concise description of the possibilities is required. This can be achieved by constructing the convex polytope associated with the LP problem at hand.

The theory of convex polytopes and normal fans is well-developed, so we will not reproduce it here. Rather we present just the definitions needed for a rigorous description of the results in Section 3. More information on these applications of polytopes is given in [4,24,31]; readers wishing to gain a thorough understanding of the theory can see [3,11,39].

The theory of linear programming starts from the following question: Given a set of feasible solutions X in $\mathbb{R}^n_{\geq 0}$ and a vector $\mathbf{A} \in \mathbb{R}^n$, for which $\mathbf{x} \in X$ is the *objective function* $\mathbf{Ax}$ minimized?

For a fixed $\mathbf{A}$, this is a straightforward question; however as $\mathbf{A}$ varies, different vectors in the set X will minimize $\mathbf{Ax}$. If X is a large or infinite set it may be simpler to deal with a geometric region containing X, than with the original set of feasible solutions itself. A *convex polytope* is a bounded region in $\mathbb{R}^n$ which can be described by a finite number of linear inequalities $\mathbf{a}_i\mathbf{x} \leq c_i$. A hyperplane is a *supporting hyperplane* for a polytope P if it intersects P non-trivially but no interior point of P is on the hyperplane. A *k-face* of P is a k-dimensional intersection of P with a supporting hyperplane. A 0-face is a *vertex* of P, a 1-face is an *edge*, and so on, with the $n-1$ dimensional faces called *facets*. The (unique) n-face is defined to be the entire polytope.

The *convex hull* of a set X is the smallest convex set containing X. If X is a finite set, its convex hull P is the set

$$P = \left\{ \sum_{\mathbf{x} \in X} b_{\mathbf{x}} \mathbf{x} : \text{ each } b_{\mathbf{x}} \geq 0 \text{ and } \sum_{\mathbf{x} \in X} b_{\mathbf{x}} = 1 \right\}. \tag{1}$$

Equivalently, Eq. (1) can be used as a definition of a convex polytope—that is, a convex polytope is the convex hull of a finite set. It is a fundamental theorem in the field that these two definitions of convex polytopes are equivalent. Figure 2 shows the convex hull of a finite set in $\mathbb{R}^2$.

The boundary of a polytope is the union of its facets and contains all faces except the n-face. If the polytope P is the convex hull of a set X of feasible solutions, then

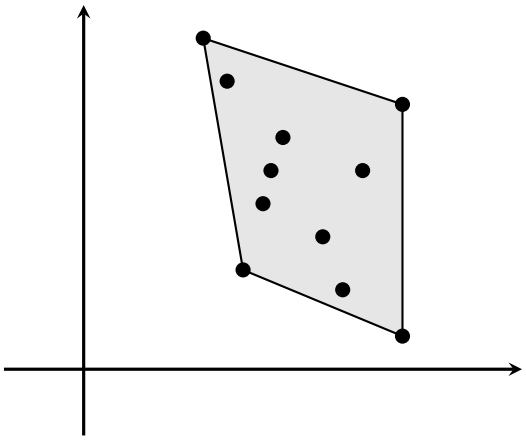


FIGURE 2. The convex hull of the set X: a polytope with four 0-faces, four 1-faces, and one 2-face.

the boundary of P contains all optimal solutions (i.e. all vectors that minimize $\mathbf{Ax}$ for non-zero $\mathbf{A}$). Thus, studying the polytope rather than the original set X simplifies the optimization problem.

In practice, however, computing the convex hull of the feasible set X is typically a hard problem. We usually know too little about the set of feasible solutions to construct the convex hull directly. In computational biology, it is usually the case that the feasible set is defined implicitly by some intrinsic rules of the linear programming problem. The real challenge, then, is to extract information about the feasible set from these intrinsic rules. In this chapter, we consider two different methods for constructing the convex hull: polytope propagation and incremental convex hull. We will discuss these methods further in Section 3.2.

The dual object to the polytope is its *normal fan*. The normal fan is a collection of regions of $\mathbb{R}^n$ called *cones*. If F is a face of the polytope, the cone corresponding to F, denoted C_F, is the set of all vectors $\mathbf{B}$ such that any point $\mathbf{y} \in F$ minimizes the product $\mathbf{Bx}$—that is,

$$C_F = \{\mathbf{B} \in \mathbb{R}^n : \mathbf{By} \leq \mathbf{Bx} \text{ for all } \mathbf{x} \in P, \mathbf{y} \in F\}.$$

The normal fan is represented visually, as in Fig. 3, by drawing the vectors $\mathbf{B} \in C_F$ with their heads at the face F.

The normal fan of a polytope of dimension d is a partition of the $\mathbb{R}^d$; every d-vector lies in some cone, and the cones are preserved by positive scalar multiplication. The cones from Fig. 3, for example, if drawn so that their heads are the origin, clearly fill the entire space $\mathbb{R}^2$, as shown in Fig. 4.

Once we have constructed the convex hull of feasible solutions for a particular LP problem, the normal fan provides ways to discuss the stability of a parameter vector $\mathbf{A}$. If $\mathbf{A}$ is near the boundary of a cone of the normal fan, then a small change in $\mathbf{A}$ will result in a different point of the polytope optimizing the function. However if $\mathbf{A}$ is far from any boundaries of its cone, then small changes will not change which point of the polytope optimizes $\mathbf{Ax}$.

In Section 3 we use the NNTM free energy calculation as our objective function and the space of all feasible secondary structures over a given RNA sequence as the

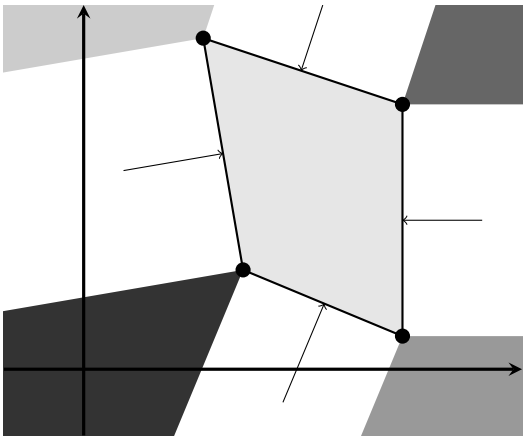


FIGURE 3. The cones corresponding to the vertices of polytope P, drawn "pointing into" their vertices. Also shown: the normal vectors to each 1-face of the polytope.

point set. Treating this objective function as a linear functional and applyling the machinery of linear programming produces what we term the *branching polytope*, which we use to analyze the parameters in the model related to multibranch loops.

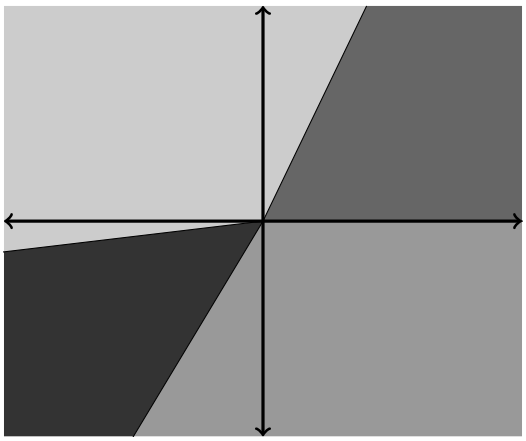


FIGURE 4. The normal fan of the polytope fills $\mathbb{R}^2$.

3. RNA BRANCHING POLYTOPE

3.1. **NNTM.** In Sections 1 and 2.1, we introduced the discrete optimization paradigm for RNA secondary structure prediction. Before we consider applications of geometric combinatorics for parametric analysis of this method, we review it in greater detail.

Discrete optimization methods for RNA secondary structure prediction derive their objective function from the NNTM. If we make the standard simplifying assumption to forbid pseudoknots (which correspond to crossings in base pairs), we

can formulate the free-energy minimization problem recursively; dynamic programming can then be used to find the minimum energy and a traceback algorithm can determine a corresponding structure.

The specific objective function used for NNTM analysis has evolved substantially over the years (cf. [22, 34, 42]), with a significant increase in the number of parameters. The Turner99 version considered here has over eight thousand parameters representing the energy contributions of various small substructures—some (~ 300) measured directly through experiments, others (~ 7000 inferred indirectly from experimental data, and finally a handful through machine learning techniques to tune the model. Among those inferred through machine learning are three connected to multibranch loops, such as the one at the center of Fig. 1.

In particular, the Turner99 version of the model we study assigns to a given multibranch loop the energy

$$\Delta G_{\text{initiation}} = a + b \cdot (\#\text{ unpaired nucleotides}) + c \cdot (\#\text{ branching helices}) \tag{2}$$

where a, b, and c are three of the learned parameters. The free energy changes associated with multibranch loops have in fact been studied experimentally [6, 18]. However, these results do not map neatly onto the linear function from Eq. (2), which is a computational simplification—a logarithmic dependence on loop size would be more biophysically accurate, but the dynamic programming approach requires a recursive decomposition that would not be possible with such a function. Recent work has led to the development of new approaches to loop energy estimation [1, 38], which rank structures more accurately but do not translate neatly into the discrete optimization setting.

Rather than focus on specific values of parameters, we investigate the behavior of the model as a function of the parameters, hoping to gain some insight into (for example) how sensitive the model is to each one and whether certain families of sequences exhibit increased or reduced sensitivity. If we could compute the collection of all secondary structures T for a given sequence S, we could attack this question directly. Unfortunately, this collection is far too large to compute explicitly—the total number of secondary structures on n bases is known to grow exponentially with n [28], and a specific sequences of biologically-relevant lengths may have 10^{50} or more possible structures—so we need to trim the problem down in scope.

To do this, we mathematically reformulate the NNTM as a simple simple linear functional focusing on the branching loop energy calculations. For a given RNA sequence, let T be a secondary structure with x multibranch loops, y unpaired nucleotides in those multibranch loops, and z helices branching from those multibranch loops. Let $a_{99} = 3.4$, $b_{99} = 0$, and $c_{99} = 0.4$ be the values of the parameters a, b, and c in the Turner99 assignment. Then the energy ΔG associated to the secondary structure T is given by

$$\Delta G_T = a_{99}x + b_{99}y + c_{99}z + w \tag{3}$$

for a "leftover" energy value w which can be computed using the model. We call the vector $\langle x, y, z, w\rangle$ the (branching) *energy signature* of the secondary structure T. Crucially, this energy signature does not actually depend on the Turner99 values of a, b, and c; the numbers x, y, and z are simply the integer counts of certain substructures, while the leftover energy w represents the non-multibranch-loop components of the energy calculation.

Now let a, b, c, and d be four arbitrary rationals. For a given secondary structure T on a given RNA sequence, we can compute an associated parameterized energy:

$$\Delta G_T(a, b, c, d) = ax + by + cz + dw \tag{4}$$

for the energy signature $\langle x, y, z, w\rangle$ of T.

Given a particular parameter vector $\langle a, b, c, d\rangle$ and RNA sequence S, we can find the secondary structure T which minimizes Eq. (4) using existing software; the details of this computation are discussed in Section 3.3. However, to understand how the model behaves for *all possible* parameter values, we will need to bring the powerful machinery of linear programming and polyhedral geometry to bear on the problem.

3.2. **Polyhedral methods and convex hulls.** Although the collection of secondary structures for a given sequence is enormous, it is nevertheless finite, and its corresponding energy signatures occupy a bounded region of $\mathbb{Q}^4$. We thus shift our attention to this region—specifically, to the *convex hull* of the collection of signature vectors for the sequence. Unlike a classical LP problem, in which the feasible region is explicitly defined by inequalities, the collection of all branching signatures is *implicitly* defined by the NNTM and the RNA sequence. As a result, we have little information about the convex hull of all branching signatures. On the other hand, we have a dynamic programming algorithm that can find an energy-minimizing RNA configuration for *any particular* set of parameters. Hence it is natural to try to construct the convex hull using this dynamic programming algorithm.

The first approach we consider is polytope propagation [23]. Polytope propagation has been used in parametric analysis of various problems, such as DNA sequence alignment and hidden Markov model for gene annotation [24]. This algorithm computes the convex hull by recursive convex hull and Minkowski sum computations on unions of polytopes. The recursions are related to the dynamic programming algorithm for calculating MFE structures. Namely, the dynamic programming algorithms we use involve decomposing a problem into a sum of smaller problems, each of which is a minimization problem involving only addition. The minimization operations can be translated into taking convex hulls of unions of polytopes, while the sums can be translated into polytope Minkowski sums, transporting the algorithm directly into the geometric domain. This translation yields a recursive representation of the RNA branching polytope.

However, while the translation from dynamic programming to polytope algebra is mathematically very elegant, polytope propagation might not be the most efficient way to compute the RNA branching polytope. The intermediate polytope computations required in this approach (convex hulls and Minkowski sum) are expensive for complicated polytopes, and these costs quickly add up. In fact, empirical results show that polytope propagation is often outperformed by incremental convex hull approaches, especially for high-dimensional models [5]. In addition, applying the polytope propagation approach would require re-implementing the optimization algorithm from scratch without taking advantage of the existing fast, peer-reviewed software for NNTM prediction.

As a result, we instead use a variant of the *Beneath-and-Beyond* approach. The core idea of Beneath-and-Beyond was introduced by Grünbaum in [11] as a method to find the facets of the convex hull of a given point set. Huggins [15] developed his `iB4e` algorithm, which applies this idea to specialized LP problems such as ours. The

basic idea of `iB4e` method is to build the convex hull incrementally—that is, to add one vertex at a time to an already constructed convex hull, by systematically solving LP problems to either expand the hull or confirm that some part of it matches the final hull. The objective vectors are chosen so that after each iteration, either a new vertex of the convex hull is found or a facet of the convex hull is confirmed. This ensures that the method requires running the LP solver for no more than $O(V + F)$ objective vectors, if the convex hull has V vertices and F facets.

Before the main loop of the algorithm can be applied, we perform an initialization step which finds a collection of points of full dimension. We then compute the convex hull of these points and label all facets of the convex hull as 'tentative', as illustrated in Fig. 5a.

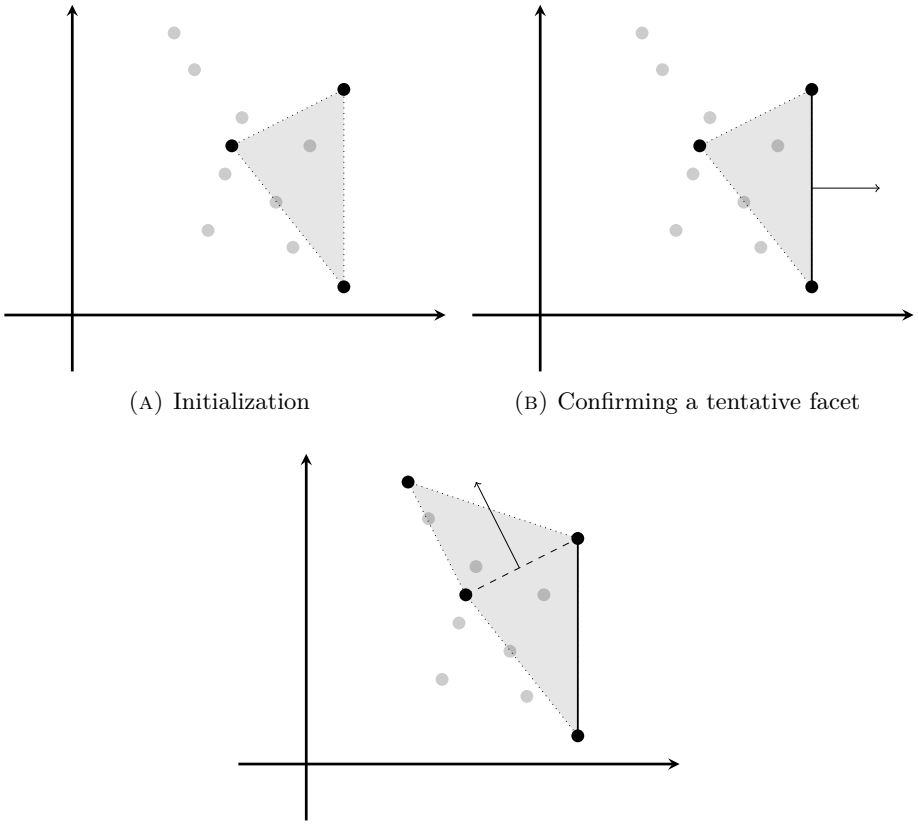
We now begin the main loop of the algorithm. At each step, we pick a tentative facet, use its outer normal vector as objective function, and solve the corresponding optimization problem. If the resulting signature is not outside the known hull, as illustrated in Fig. 5b, that facet becomes 'confirmed' and the process is restarted with another tentative facet. However, if the resulting signature is outside the known hull, as illustrated in Fig. 5c, we add it to the vertex set, compute the new convex hull, and label the newly added facets as tentative.

The process is repeated until all facets are confirmed. Since the point set is finite, the process must terminate, and the end result is the convex hull of the set of *all* solutions to the LP solver. In our case, this is the branching polytope, and once we have computed it we can easily compute its normal fan using standard methods and proceed to study the model.

3.3. **Computing the branching polytope with `pmfe`.** It is conceptually natural to apply the Beneath-and-Beyond algorithm to compute the branching polytope of a particular RNA sequence by treating the parametrized NNTM minimization problem as a linear program. However, coupling the two algorithms to perform the computation requires some care. We introduce the software `pmfe`, available publicly at `https://github.com/AMS-MRC-disc-math-bio/pmfe`. Instructions to build and install the software are provided in the file `README.md` of that repository. Alternatively, a ready-to-use version of the software is available in a Docker container, posted publicly on the Docker hub as `agdphd/pmfe`.

The `pmfe` package has three components (see Fig. 6): `iB4e`, `findmfe`, and `scorer`.

- `iB4e` is an implementation of Huggins' algorithm as a header-only template library in C++, taking advantage of the CGAL computational geometry library [30] to handle the geometric computations.
- `findmfe` takes a parameter vector $\mathbf{P} = \langle a, b, c, d\rangle$ and an RNA sequence S as input and returns as output a secondary structure on S which has minimal energy with respect to $\mathbf{P}$. From the available software packages which implement NNTM optimization, we chose to base `findmfe` on `GTfold` [20] because it is parallelized and thus able to take advantage of modern multi-core desktop computers. Some modifications were required to adapt it to our use. For computational efficiency, `GTfold` performs all arithmetic with fixed precision of two decimal places; we converted it to use arbitrary-precision rational arithmetic based on the GMP library [10], which increases running time by an order of magnitude but ensures that the results are exact even when parameters are not multiples of 1/100. We also added an interface to modify the multibranch parameters $\langle a, b, c\rangle$ and the dummy



(A) Initialization

(B) Confirming a tentative facet

(C) Extending a tentative facet

FIGURE 5. The Beneath-and-Beyond algorithm for polytope construction

scaling parameter d programmatically, allowing the Beneath-and-Beyond algorithm to treat `GTfold` as a function of these four variables and call it repeatedly.

- `scorer` takes an RNA sequence S and a secondary structure T and returns the signature vector $\langle x, y, z, w \rangle$ of T according to Eq. (3). (In particular, it computes the value of w by computing the energy decomposition of T under the Turner99 parameters.)

We combine these three components with the control flow illustrated in Fig. 6 to create the `pmfe` software package. Given an RNA sequence S, the `iB4e` module repeatedly generates parameter vectors according to Huggins' algorithm. Each of these vectors is sent to the `findmfe` module as a query, which generates a secondary structure with minimal energy. Each such secondary structure is sent to the `scorer` module to find its signature vector, which is then sent back to `iB4e` as the response to the query. After some number of iterations of this loop, the convex hull of these signature points s is found to be equal to the branching polytope, which is returned to the user. (For convenience, we in fact provide the user with a file containing both the signatures s and their associated structures T in condensed dot-bracket notation.)

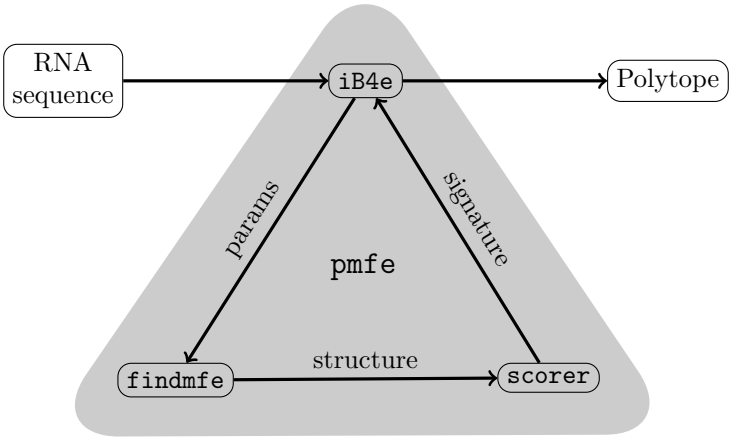


FIGURE 6. Control flow of the `pmfe` software

Since the software `pmfe` which we introduce is complex and extensively modifies the peer-reviewed codebase of `GTfold`, we provide a testing framework to ensure correctness. Tests are available to ensure that `pmfe` returns the same structures and scores as `GTfold`, using both the classical Turner99 parameters [17] and various modified parameter sets for a variety of natural and synthetic RNA sequences. Details for running these tests are available in the `pmfe` repository.

We also provide a module `rna_poly` in the SageMath computer algebra system [29] which can be used to study the branching polytopes produced by `pmfe`. This can be used to generate a variety of visualizations of the normal fan of the branching polytope. In particular, it provides a simple interface for taking the $d = 1$ slice of the fan, giving a polyhedral partition of $\mathbf{R}^3$ whose regions are collections of parameters $\langle a, b, c\rangle$ which yield the same secondary structure for the sequence under study. The results are illustrated in Fig. 7, which shows the $b = 0, d = 1$ slice of the normal fan of the branching polytope for a *H.sapiens* tRNA sequence. The Turner99 parameter values $a_{99} = 3.4, c_{99} = 0.4$ are marked with a circle just northeast of the origin.

3.4. **Biological questions leading to mathematical problems.** The RNA branching polytope and the associated polyhedral decomposition of the three-dimensional multibranch parameter space obtained by intersecting its normal fan with $d = 1$ encapsulate the dependency of the optimization on the parameters a, b, c. Various questions related to accuracy and stability can be addressed through the analysis of the polytope and the polyhedral decomposition. However, answering such questions in a manner that is biologically relevant requires consideration of certain subtleties and this leads to interesting mathematical problems. Here we discuss a few biological questions that could be addressed through the parametric analysis and related mathematical problems that arise.

Question 1: *How robust is the optimization for a given RNA sequence?*

In the strictest sense, the prediction is sensitive if a variation of the parameters within an allowed margin of error produces different optimal structures. This occurs if the distance of the Turner99 parameters (a_{99}, b_{99}, c_{99}) from the boundary of the polyhedron that contains this point is smaller than the allowed margin of

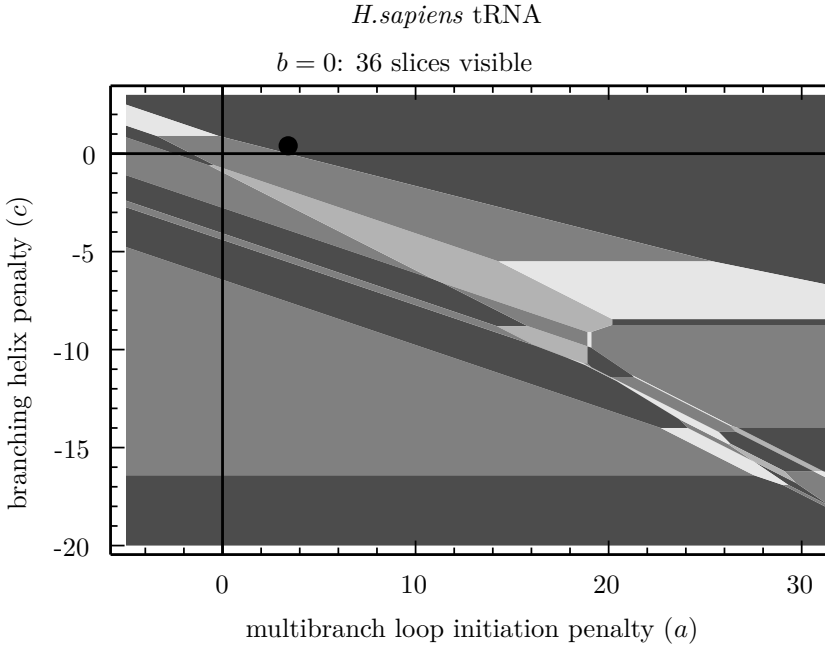


FIGURE 7. The intersection of the parameter space decomposition for a *H.sapiens* tRNA sequence with $b = 0$.

error. For example, Fig. 7 shows the position of (a_{99}, b_{99}, c_{99}) within the polyhedral decomposition of a *H.sapiens* tRNA (only the $b = 0$ slice is depicted, which is the baseline value in Turner99). Its distance to the boundary is 0.13699 and, for instance, optimization using $a = 3.39130, b = -0.14786, c = 0.37391$ yields structures with a different signature. However, this by itself does not mean that the optimization is very sensitive; analysis of the structure space is required so that one can quantify the sensitivity. Namely, two structures with different signatures may still have a lot of structural similarities (for example, very similar long helices and branching pattern), in which case one would not necessarily say that the prediction is very sensitive. Moreover, since the structure-to-signature map is not one-to-one, when quantifying the structural changes in the optimal structures one actually needs to compare two sets of structures, each one corresponding to a given signature. In short, in the assessment of robustness we face the following problem.

Problem 1: *Define a good representative of the set of structures that correspond to the same signature.*

The problem of representing a whole set of structures by a single one has appeared before, for example in the context of compact representation of a sample of structures. However, the methods developed for this (e.g. sfold clustering [7] and profiling [27]) usually assume structural similarities of the elements in the set. Here, though, we are presented with a different set-up: the structures mapping to the same signature do not a priori have any common motifs. Therefore, choosing one consensus structure as the other methods do may not yield a reasonable representative of the whole set.

Question 2: *How much can the prediction be improved for a given sequence?*

For sequences for which the native structure has been obtained experimentally or via comparative sequence analysis, one could assess the accuracy of the NNTM by

comparing it to the MFE structure. In fact, since the structures corresponding to all signatures on the boundary of the branching polytope can in practice be obtained fairly efficiently [37], one can precisely determine branching parameters that would yield a signature corresponding to a structure that is closest to the native one. However, here we again face the problem that besides the most accurate one, that signature corresponds to a whole set of MFE structures that might be very different from the native structure. In such a case, the almost accurate structure might be unrecognizable. Therefore, in assessing accuracy, the problem of finding a good representative structure for a given signature is still very relevant. Furthermore, even if we can solve this problem, the most accurate prediction obtained this way may require a triple of parameters (a, b, c) that is very different from (a_{99}, b_{99}, c_{99}), which may not be very desirable. Namely, even though (a_{99}, b_{99}, c_{99}) were not obtained experimentally, they still might have some biological relevance since the initiation point for the genetic algorithm used was suggested by stabilities determined by optical melting for an RNA multibranch loop with three branching helices [17]. This suggests the following natural problem.

Problem 2: *Maximize the improvement in accuracy while minimizing parameter change.*

The first mathematical subproblem here is to formulate an objective function which takes into account appropriately weighted structural and parameter changes. While there are simple structure metrics that can be very efficiently computed (e.g. the base pair metric based on symmetric set difference), other metrics may be more appropriate to quantify the structural differences [21]. Deciding how to weight the multibranch parameters, however, seems to be an even more challenging problem since it is not clear how the parameter changes are related to the structural changes on average.

Question 3: *Is there a choice of branching parameters that would improve the prediction accuracy for a family of sequences?*

Homologous sequences perform the same function in different species and they fold into essentially the same structure. However, their MFE prediction accuracies can vary significantly. For instance, within the tRNA family, the accuracy calculated as the F-measure (the harmonic mean of the MFE sensitivity and positive predictive value against true positive base pairs) ranges from the minimal 0 to the maximal 1 [27]. Hence, it is natural to try to find a set of parameters which yields the highest accuracy for a family of sequences. For this, one needs to address the question: *Are there any patterns in the dependency of the prediction on the parameters across a family of homologous sequences?* Mathematically, one could phrase this as a problem of finding similarities between polytopes in $\mathbb{R}^4$ or between polyhedral decompositions of $\mathbb{R}^3$. For example, Fig. 8 shows the $b = 0$ slice of the polyhedral decompositions for a tRNA sequence of *L.delbrueckii*. By visually comparing this figure to Fig. 7 one can notice similarities in the directions of the splitting hyperplanes. The challenge of course is to quantify such similarities. So, we face the following problem.

Problem 3: *Find a way to compare branching polytopes.*

Comparison of polytopes is a problem that has appeared before, for instance in relation to the traveling salesman problem in optimization [9]. Even in optimization problems when one works with relaxation polytopes that are, loosely speaking, "close", comparing relaxations is not well understood and different comparison methods have been suggested (e.g. [16]). In our case, an additional challenge is the

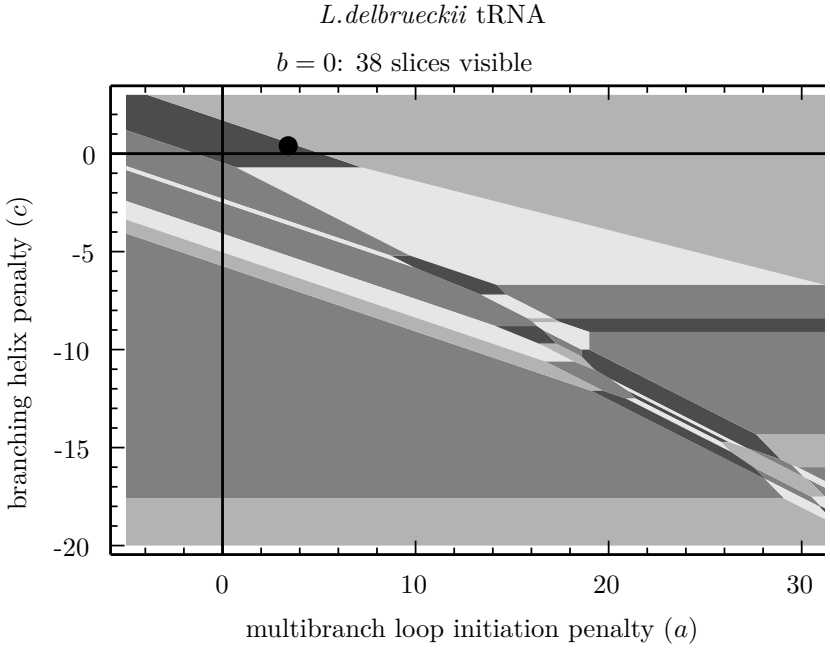


FIGURE 8. The $b = 0$ slice of the parameter space decomposition for a *L.delbrueckii* tRNA sequence.

fact that the polytopes to be compared do not a priori satisfy any mathematically tractable assumptions for closeness.

4. Conclusions

The affine energy function which governs the branching of an RNA secondary structure under the NNTM optimization is a known weakness of the thermodynamic model, and the methods from geometric combinatorics outlined here offer significant potential to assess its accuracy and stability through a parametric analysis. While simplified models are certainly more tractable, the new `pmfe` computational framework makes possible the analysis of branching polytopes for real RNA sequences for the first time. As illustrated by the qualitative similarities between Fig. 7 and Fig. 8, this approach is capturing some interesting characteristics which are preserved between two different sequences. Moreover, as discussed, moving beyond the qualitative comparison of polytope slices to quantify their similarities and differences presents some interesting mathematical challenges. Consequently, it remains to be seen whether the observed patterns in these proof-of-principle results are resulting from the structure of the thermodynamic optimization and/or the coding of structural motifs in the base pairing of these RNA sequences. Resolving this dichotomy will reveal much about the interplay between mathematics and biology in the prediction of RNA secondary structures.

References

[1] Daniel P Aalberts and Nagarajan Nandagopal, *A two-length-scale polymer theory for RNA loop free energies and helix stacking*, RNA **16** (2010), no. 7, 1350–1355.

[2] Andrew D Bates and Anthony Maxwell, *DNA topology*, Oxford University Press, Oxford, UK, 2005.
[3] Dimitris Bertsimas and John N Tsitsiklis, *Introduction to linear optimization*, Athena Scientific Series in Optimization and Neural Computation, Athena Scientific, 1997.
[4] Jesús De Loera, Raymond Hemmecke, and Matthias Köeppe, *Algebraic and geometric ideas in the theory of discrete optimization*, MOS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2012.
[5] Colin N Dewey, Peter M Huggins, Kevin Woods, Bernd Sturmfels, and Lior Pachter, *Parametric alignment of Drosophila genomes*, PLoS Comput Biol **2** (2006), no. 6, e73.
[6] Joshua M Diamond, Douglas H Turner, and David H Mathews, *Thermodynamics of three-way multibranch loops in RNA*, Biochemistry **40** (2001), no. 23, 6971–6981.
[7] Ye Ding, Chi Yu Chan, and Charles E Lawrence, *RNA secondary structure prediction by centroids in a Boltzmann weighted ensemble*, RNA **11** (2005), no. 8, 1157–1166.
[8] Richard Durbin, Sean R Eddy, Anders Krogh, and Graeme Mitchison, *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*, Cambridge University Press, 1998.
[9] Michel X Goemans, *Worst-case comparison of valid inequalities for the TSP*, Mathematical Programming **69** (1995), no. 1-3, 335–349.
[10] Torbjörn Granlund and the GMP development team, *GNU MP: The GNU Multiple Precision Arithmetic Library*, 6.0.0a, 2014. `http://gmplib.org/`.
[11] Branko Grünbaum, *Convex polytopes*, Graduate Texts in Mathematics, vol. 221, Springer-Verlag, New York, 2003.
[12] Dan Gusfield, *Algorithms on strings, trees and sequences: computer science and computational biology*, Cambridge University Press, 1997.
[13] Ivo L Hofacker, Walter Fontana, Peter F Stadler, L Sebastian Bonhoeffer, Manfred Tacker, and Peter Schuster, *Fast folding and comparison of RNA secondary structures*, Monatshefte für Chemie/Chemical Monthly **125** (1994), no. 2, 167–188.
[14] Valerie Hower and Christine E Heitsch, *Parametric analysis of RNA branching configurations*, Bulletin of Mathematical Biology **73** (2011), no. 4, 754–776.
[15] Peter Huggins, *iB4e: A software framework for parametrizing specialized LP problems*, Mathematical software–icms 2006, 2006, pp. 245–247.
[16] Jon Lee and Walter D Morris, *Geometric comparison of combinatorial polytopes*, Discrete Applied Mathematics **55** (1994), no. 2, 163–182.
[17] 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** (1999), no. 5, 911–940.
[18] David H Mathews and Douglas H Turner, *Experimentally derived nearest-neighbor parameters for the stability of RNA three- and four-way multibranch loops*, Biochemistry **41** (2002), no. 3, 869–80.
[19] ———, *Prediction of RNA secondary structure by free energy minimization*, Current Opinion in Structural Biology **16** (2006), no. 3, 270–278.
[20] Amrita Mathuriya, David A Bader, Christine E Heitsch, and Stephen C Harvey, *GTfold: a scalable multicore code for RNA secondary structure prediction*, Proceedings of the 2009 acm symposium on applied computing, 2009, pp. 981–988.
[21] Vincent Moulton, Michael Zuker, Michael Steel, Robin Pointon, and David Penny, *Metrics on RNA secondary structures*, Journal of Computational Biology **7** (2000), no. 1-2, 277–292.
[22] Ruth Nussinov, George Pieczenik, Jerrold R Griggs, and Daniel J Kleitman, *Algorithms for loop matchings*, SIAM Journal on Applied mathematics **35** (1978), no. 1, 68–82.
[23] Lior Pachter and Bernd Sturmfels, *Parametric inference for biological sequence analysis*, Proceedings of the National Academy of Sciences of the United States of America **101** (2004), no. 46, 16138–16143.
[24] ———, *Algebraic statistics for computational biology*, Vol. 13, Cambridge University Press, 2005.
[25] Pavel Pevzner, *Computational molecular biology: an algorithmic approach*, MIT Press, 2000.
[26] Jessica S Reuter and David H Mathews, *RNAstructure: software for RNA secondary structure prediction and analysis*, BMC bioinformatics **11** (2010), no. 1, 129.
[27] Emily Rogers and Christine E Heitsch, *Profiling small RNA reveals multimodal substructural signals in a Boltzmann ensemble*, Nucleic Acids Research (2014).

[28] PR Stein and MS Waterman, *On some new sequences generalizing the Catalan and Motzkin numbers*, Discrete Mathematics **26** (1979), no. 3, 261–272.
[29] W. A. Stein et al., *Sage Mathematics Software (Version 6.7)*, The Sage Development Team, 2015. `http://www.sagemath.org`.
[30] The CGAL Project, *CGAL user and reference manual*, 4.6, CGAL Editorial Board, 2015.
[31] Rekha R Thomas, *Lectures in geometric combinatorics*, Vol. 33, American Mathematical Society, 2006.
[32] Ignacio Tinoco and Carlos Bustamante, *How RNA folds*, Journal of Molecular Biology **293** (1999), no. 2, 271–281.
[33] Douglas H Turner and David H Mathews, *NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure*, Nucleic Acids Research (2009), gkp892.
[34] Michael Waterman, *Secondary structure of single-stranded nucleic acids*, Studies on foundations and combinatorics, 1978, pp. 167–212.
[35] Michael S Waterman, *Parametric and ensemble sequence alignment algorithms*, Bulletin of Mathematical biology **56** (1994), no. 4, 743–767.
[36] ———, *Introduction to computational biology: Maps, sequences and genomes*, CRC Press, 1995.
[37] Stefan Wuchty, Walter Fontana, Ivo L Hofacker, Peter Schuster, et al., *Complete suboptimal folding of RNA and the stability of secondary structures*, Biopolymers **49** (1999), no. 2, 145–165.
[38] Jian Zhang, Ming Lin, Rong Chen, Wei Wang, and Jie Liang, *Discrete state model and accurate estimation of loop entropy of RNA secondary structures*, The Journal of chemical physics **128** (2008), no. 12, 125107.
[39] Günter M Ziegler, *Lectures on polytopes*, Graduate Texts in Mathematics, Springer-Verlag, 1995.
[40] Michael Zuker, *RNA folding prediction: The continued need for interaction between biologists and mathematicians*, Some mathematical questions in biology: DNA sequence analysis, 1986, pp. 87–124.
[41] ———, *On finding all suboptimal foldings of an RNA molecule*, Science **244** (1989), no. 4900, 48–52.
[42] Michael Zuker and Patrick Stiegler, *Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information*, Nucleic Acids Research **9** (1981), no. 1, 133–148.

Department of Mathematics, University of North Texas, Denton, TX, USA 76203
E-mail address: `elizabeth.drellich@unt.edu`

Department of Mathematics and Computer Science, Hobart and William Smith Colleges, Geneva, New York, USA 14456
Current address: Center for Quantitative Medicine, UConn Health, Farmington, CT, USA 06051
E-mail address: `andrew.gainer.dewar@gmail.com`

Mathematical Institute, University of Oxford, Andrew Wiles Building, Radcliffe Observatory Quarter, Woodstock Road, Oxford OX2 6GG United Kingdom
E-mail address: `harrington@maths.ox.ac.uk`

Department of Mathematical Sciences, Clemson University, Clemson, SC, USA 29634
E-mail address: `qhe@clemson.edu`

School of Mathematics, Georgia Institute of Technology, Atlanta, GA, USA 30332
E-mail address: `heitsch@math.gatech.edu`

Department of Mathematical Sciences, Clemson University, Clemson, SC, USA 29634
E-mail address: `spoznan@clemson.edu`