BPPart: RNA-RNA Interaction Partition Function in the Absence of Entropy

Ali Ebrahimpour-Boroojeny ✉
Department of Computer Science, Columbia University, New York, NY, USA [1]
New York Genome Center, NY, USA

Sanjay Rajopadhye ✉
Department of Computer Science, Colorado State University, Fort Collins, CO, USA

Hamidreza Chitsaz ✉
Waymo, Mountain View, CA, USA[2]

Abstract

A few classes of RNA-RNA interaction (RRI) with complex roles in cellular functions, such as miRNA-target and lncRNAs, have already been studied. Accordingly, RRI bioinformatics tools proposed in the last decade are tailored for those specific classes. Interestingly, there are somewhat unnoticed mRNA-mRNA interactions in the literature with potentially drastic biological roles. Hence, there is a need for high-throughput *generic* RRI bioinformatics tools that can be used in more comprehensive settings. In this work, we revisit two of the RRI partition function algorithms, `piRNA` and `rip`. These are equivalent methods that implement the most comprehensive and computationally intensive thermodynamic model for RRI. We propose simpler models that are shown to retain the vast majority of the thermodynamic information that the more complex models capture. Specifically, we simplify the energy model by ignoring the system's entropy and show its equivalency to a base-pair counting model. We allow different weights for base-pairs to maximize the correlations with the full thermodynamic model. Our newly developed algorithm, `BPPart`, is 225× faster than `piRNA` and is more expressive and easier to analyze due to its simplicity and order of magnitude reduction in the number of dynamic programming tables. Still, based on our analysis of both the real and randomly generated data, its scores achieve a correlation of 0.855 with `piRNA` at $37°C$. Finally, we illustrate one use-case of such simpler models to generate hypotheses about the roles of specific RNAs in various diseases. We have made our tool publicly available and believe that this faster and more expressive model will make the incorporation of physics-guided information in complex RRI analysis and prediction models more accessible.

2012 ACM Subject Classification Applied computing → Computational biology

Keywords and phrases RNA-RNA Interaction, Partition Function, RNA Secondary Structure

Digital Object Identifier 10.4230/LIPIcs.WABI.2021.14

Related Version *Full Version*: `https://arxiv.org/abs/1904.01235`

Supplementary Material *Software (Source Code)*: `https://github.com/Ali-E/bipart`
archived at `swh:1:dir:e8583c3964d5a1f404e893e461b70c2ece40c692`

1 Introduction

Since mid 1990s with the advent of RNA interference discovery, RNA-RNA interaction (RRI) has moved to the spotlight in modern, post-genome biology. RRI is ubiquitous and has increasingly complex roles in cellular functions. In human health studies, miRNA-target and lncRNAs are among an elite class of RRIs that have been extensively studied and shown to play significant roles in various diseases including cancer. Bacterial ncRNA-target and

[1] Work was done when the author was at Colorado State University.
[2] Work was done when the author was on the faculty of Computer Science at Colorado State University.

21st International Workshop on Algorithms in Bioinformatics (WABI 2021).
Editors: Alessandra Carbone and Mohammed El-Kebir; Article No. 14; pp. 14:1–14:24
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

RNA interference are other classes of RRIs that have received significant attention. However, new evidence suggests that other classes of RRI, such as mRNA-mRNA interactions, are biologically important. The RISE database [16] reports a number of biologically significant instances of mRNA-mRNA interactions. These representative mRNA-mRNA interactions suggest that general RRIs, including mRNA-mRNA interactions, play major roles in human biology. Hence, there is a need for high-throughput *generic* RNA-RNA interaction bioinformatic tools for all types of RNAs.

In this paper, we revisit the well-studied problem of RNA-RNA interaction, and investigate the trade-off of complexity of the full thermodynamic models, such as piRNA [8] and rip [21], and accuracy of the scores they can generate. The aforementioned models are computationally intensive, and this prohibits their application to not only large-scale studies, but even for average sized pairs of RNAs. Because of the equivalency of these models, and availability of piRNA (the links to the tool provided by Huang et al. [21] are broken), we chose piRNA as the representative of the two in our experiments and analysis. piRNA is a dynamic programming algorithm that computes the partition function, base-pairing probabilities, and structure for the comprehensive Turner energy model in $O(n^4m^2 + n^2m^4)$ time and $O(n^4 + m^4)$ space. Due to intricacies of the energy model, including various (kissing) loops such as hairpin loop, bulge/internal loop, and multibranch loop, piRNA involves 96 different dynamic programming tables and needs multiple table look-ups for computing their values. An implementation of piRNA is currently available at http://chitsazlab.org/software/pirna.

In this paper, we introduce a strategic retreat from the slower comprehensive models such as piRNA by simplifying the energy model; we ignore the systems' entropy and derive a model that only requires the consideration of simple weighted base-pair counting. We develop the BPPart algorithm which aims to solve this simpler model with a much simpler approach. We also allow different weights for base-pairs which helps us to attain a model which correlates well with the full thermodynamic ones. In addition to this algorithm, we implemented a correct version of an earlier developed method, IRIS [39], which is based on base-pair maximization criterion, to have a thorough comparison between all these methods which are vastly different in terms of complexity. The implementation of this model, which we named BPMax, is also available in our publicly-available repository, and the results related to that are available in the Supplementary Material.

By the explosion of experimental data and the necessity to have higher-throughput methods, this retreat seems necessary, especially if one is willing to have more expressive models or wants to build physics-guided models that retain most of the information that can be derived from the thermodynamic system of RRI. BPPart involves eight 4-dimensional dynamic programming tables, and BPMax involves only one 4-dimensional table. Both BPPart and BPMax compared with piRNA are simpler dynamic programming algorithms which are more than 225× and 1300× faster, respectively, on the 50,500 RRI samples we used for our experiments. The reason for this noticeable speed-up is reducing the number of tables and the number of table look-ups for computing the new values and also the fact that the 96 large tables of piRNA renders piRNA memory- rather than compute-bound in practice. Moreover, the significantly reduced memory footprint of BPPart and BPMax makes them feasible targets for optimization on different hardware platforms like GPU based accelerators, an avenue we plan to explore in the future.

The key question concerns the accuracy we lose by simplifying the scoring model from the comprehensive Turner model to simply weighted base-pair counting. We answer this by computing both the Pearson and Spearman's rank correlations at different temperatures between the results of BPPart, BPMax, and piRNA on 50,500 experimentally characterized

RRIs in the RISE database [16]. We find that the Pearson correlations between BPPart and piRNA is 0.855 and BPMax and piRNA is 0.836 at $37°C$. Based on the results, we conclude that BPPart and BPMax capture a significant portion of the thermodynamic information. The simpler and faster algorithms, allow them to be used in high-throughput methods and be complemented with machine learning techniques in the future for more accurate predictions.

1.1 Related work

During the last few decades, several computational methods emerged to study the secondary structure of single and interacting nucleic acid strands. Most use a thermodynamic model such as the well-known Nearest Neighbor Thermodynamic model [32, 6, 13, 8, 38, 50, 54, 44, 33, 51]. Some previous attempts to analyze the thermodynamics of multiple interacting strands concatenate input sequences in some order and consider them as a single strand [2, 3, 12]. Alternatively, several methods avoid internal base-pairing in either strand and compute the minimum free energy secondary structure for their hybridization under this constraint [42, 11, 31]. The most comprehensive solution is computing the joint structure between two interacting strands under energy models with a growing complexity [40, 1, 29, 10, 23, 8, 21].

Other methods predict the secondary structure of individual RNA independently, and predict the (most likely) hybridization between the unpaired regions of the two interacting molecules as a multistep process: 1) unfolding of the two molecules to expose bases needed for hybridization, 2) the hybridization at the binding site, and 3) restructuring of the complex to a new minimum free energy conformation [35, 49, 5, 7]. The success of such methods, including our biRNA algorithm [7], suggests that the thermodynamic information vested in subsequences and pairs of subsequences of the input RNAs can provide valuable information for predicting features of the entire interaction.

In addition to general RNA-RNA interaction tools, many tools have been developed to predict the secondary structure of interacting RNAs for a specific type of interest which has been shown to be more effective in some cases due to the utilization of certain properties belonging to that type. As mentioned earlier, miRNA-target prediction is one such class of high interest for which such specialized tools have been created to incorporate various properties specific to miRNAs; some of these tools use the seed region of a miRNA which is highly conserved [26, 25, 27, 53], some consider the free energy to compute accessibility to the binding site in $3'$ UTR [18, 29, 25], some utilize the conservation level which is derived using the phylogenetic distance [36, 4, 41, 15, 26, 25], and some others consider other target sites as well, such as the $5'$ UTR, Open Reading Frames (ORF), and the coding sequence (CDS) for mRNAs [43, 34, 19, 52].

There are also several other tools developed for other specific types of RNA; IntaRNA [5, 30] is one such tool that although is used for RNA-RNA interaction in general, it is primarily designed for predicting target sites of non-coding RNAs (ncRNAs) on mRNAs. There are many other examples, such as PLEXY [24] which is a tool designed for C/D snoRNAs, RNAsnoop [45] that is designed for H/ACA snoRNAs, TargetRNA [46] which is a tool aimed at predicting interaction of bacterial sRNAs [48].

2 MATERIALS AND METHODS

Here we describe how our algorithm, BPPart, utilizes a dynamic programming approach to compute the partition function for RNA-RNA interaction when entropy is ignored and only a weighted score for pairing different nucleotides is considered. This algorithm is guaranteed to be mutually exclusive on the set of structures, i.e., it counts each structure exactly once.

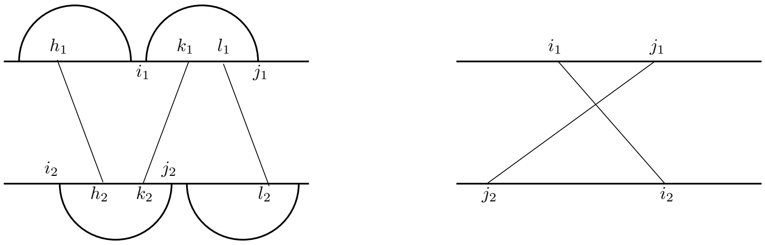

Figure 1 An illustration of a zigzag (left) and a crossing bond (right), which are excluded in our algorithm.

For `BPMax` which maximizes the (weighted scores) of base-pairs, such mutual exclusion is not necessary because the max operator is idempotent (counting the same structure multiple times does not affect the value of the objective function) and we can derive a simpler recursion. Our codes are freely available under open source license.

Preliminaries

In this paper, we mostly follow the notations and definitions used to develop our `piRNA` algorithm [8]. We denote the two nucleic acid strands by $\mathbf{R}$ and $\mathbf{S}$. Strand $\mathbf{R}$ is indexed from 1 to L_R, and $\mathbf{S}$ is indexed from 1 to L_S both in $5'$ to $3'$ direction. Note that the two strands interact in opposite directions, e.g. $\mathbf{R}$ in $5' \to 3'$ with $\mathbf{S}$ in $3' \leftarrow 5'$ direction; however, we consider the reverse of $\mathbf{S}$ in our figures for clearer illustration of the configurations. Each nucleotide is paired with at most one nucleotide in the same or the other strand. The subsequence from the i^{th} nucleotide to the j^{th} nucleotide, inclusive, in either strand is denoted by $[i, j]$.

An intramolecular base pair between the nucleotides i and j (by convention, $i < j$) in a strand is called an *arc* and denoted by a bullet $i \bullet j$. We represent the score of such arc by $\text{score}(i, j)$. Essentially, $\text{score}(i, j)$ is c_1 if $i \bullet j$ is GU or UG, is c_2 if $i \bullet j$ is AU or UA, and is c_3 if $i \bullet j$ is CG or GC. An intermolecular base pair between the nucleotides k_1 and k_2, where $k_1 \in R, k_2 \in S$, is called a *bond*, denoted by a circle $k_1 \circ k_2$. We represent the score of such a bond by $\text{iscore}(k_1, k_2)$. Essentially, $\text{iscore}(k_1, k_2)$ is c'_1 if $k_1 \circ k_2$ is GU or UG, is c'_2 if $k_1 \circ k_2$ is AU or UA, and is c'_3 if $k_1 \circ k_2$ is CG or GC.

An arc $i \bullet j$ in R *covers* a bond $k_1 \circ k_2$ if $i_1 < k_1 < j_1$. We call $i \bullet j$ an *interaction arc* in R if there is a bond $k_1 \circ k_2$ covered by $i \bullet j$. The *scope* of an interaction arc is the interval $[i+1, j-1]$. We call a base on either strand an *event* if it is either the end point of a bond or that of an interaction arc. In our explanation we may use arc and bond as verbs. Two bonds $i_1 \circ i_2$ and $j_1 \circ j_2$ are called *crossing bonds* (right case of Figure 1) if $i_1 < j_1$ and $i_2 > j_2$, or vice versa. An interaction arc $i_1 \bullet j_1$ in a strand *subsumes* a subsequence $[i_2, j_2]$ in the other strand if none of the bases in $[i_2, j_2]$ has a bond with a base outside this arc. Mathematically, for all bonds $k_1 \circ k_2$ where $i_2 < k_2 < j_2$, k_1 lies within the scope of $i_1 \bullet j_1$. Two interaction arcs are *equivalent* if they subsume one another. Two interaction arcs $i_1 \bullet j_1$ and $i_2 \bullet j_2$ are part of a *zigzag*, if neither $i_1 \bullet j_1$ subsumes $[i_2, j_2]$ nor $i_2 \bullet j_2$ subsumes $[i_1, j_1]$ (left case of Figure 1).

In this work, we assume there are no pseudoknots in individual secondary structures of $\mathbf{R}$ and $\mathbf{S}$, and also there are no crossing bonds and no zigzags between $\mathbf{R}$ and $\mathbf{S}$. These constraints allow a polynomial algorithm – the general case of considering all possible structures is NP-hard [1]. We denote the ensemble of unpseudoknotted structures of $\mathbf{R}$ and $\mathbf{S}$ by $\mathcal{S}(\mathbf{R})$ and $\mathcal{S}(\mathbf{S})$ respectively. The ensemble of unpseudoknotted, crossing-free, and zigzag-free joint interaction structures is denoted by $\mathcal{S}^I(\mathbf{R}, \mathbf{S})$.

For a given joint interaction structure $s \in \mathcal{S}^I(\mathbf{R}, \mathbf{S})$, let AU$(s)$ denote the number of A-U base pairs in s. Similarly, CG(s) and GU(s) denote the number of C-G and G-U base pairs in s, respectively. We define *bpcount* as a weighted sum, for some constants, $c_1, \ldots, c_3$

$$\text{bpcount}(s) = c_1\text{GU}(s) + c_2\text{AU}(s) + c_3\text{CG}(s). \tag{1}$$

Rivas-Eddy Diagrams

For the sake of completeness, we describe the "Rivas-Eddy diagram" notation that we adopt in this paper in the Supplementary Material. The Rivas-Eddy diagram to compute a certain function is written like a formal (context free) grammar. The left hand side is labeled with the name of a table (structure), and the right hand side has a number of alternate substructures separated by vertical bars. Often, some of the boundary cases (e.g., singleton or empty subsequences) are omitted for brevity.

2.1 Problem Definition

The Gibbs free energy

$$\Delta G = \Delta H - T\Delta S \tag{2}$$

is composed of a term ΔH called enthalpy that does not depend on temperature and a term $T\Delta S$ that includes entropy and is linearly dependent on temperature T. Intuitively, enthalpy is the chemical energy that is often released upon formation of chemical bonds such as base pairing. Entropy, on the other hand, captures the size of all possible spatial conformations for a fixed secondary structure. In other words, entropy captures the amount of 3D freedom of the molecule. A base-pair brings enthalpy down, hence favorable from an enthalpy point of view, and decreases freedom (entropy), hence unfavorable from an entropy point of view. These two opposing objectives are combined linearly through the temperature coefficient.

In the full thermodynamic model, we consider both terms. In the base pair counting, we consider only a simplistic enthalpy term. Partition function for the full thermodynamic model is

$$\sum_{s \in \mathcal{S}^I} e^{-\Delta G/RT}, \tag{3}$$

in which R is the gas constant, and S^I are all possible states of the system, assuming that they form a countable set (which they do in our case by we considering all possible ways the two RNAs pair with one another). Now, by ignoring the term with the entropy, and considering the approximation $\Delta G \sim \Delta H$, we can simplify the model as follows

$$\sum_{s \in \mathcal{S}^I} e^{-\Delta G/RT} \approx \sum_{s \in \mathcal{S}^I} e^{-\Delta H/RT} \approx \sum_{s \in \text{S}^I(\mathbf{R},\mathbf{S})} e^{-\text{bpcount}(s)/RT}. \tag{4}$$

To make the 3rd term a better approximation for the first one, we allow different weights for different base-pairs (`AC`, `GT`, and `CG`) in our model. We optimize these weights to maximize the correlation of the scores with those of `piRNA` (which is based on the first term above) and verify the consistency of the computed weights using a randomly generated dataset. So, basically, by allowing the base-pairs to have different weights and finding the optimum ones, we seek to minimize the information we lose by ignoring the term with the entropy.

In our experiments, we also perform analysis on the base-pair maximization model, `BPMax`, which finds the structure that has the maximum weighted base pair count, i.e.

$$\mathrm{BPMax}(\mathbf{R},\mathbf{S}) = \max_{s\in \mathrm{S}^I(\mathbf{R},\mathbf{S})} \mathrm{bpcount}(s). \tag{5}$$

This problem (without the weights for base-pairs) was previously studied by Pervouchine [40] in an algorithm called `IRIS`. However, there is no publicly available correct implementation of `IRIS`. As in `BPPart`, we allow weighted scores for the base-pairs in the `BPMax` algorithm to maximize the correlation of its scores with those of `piRNA`. We give a dynamic programming algorithm for this model in the Supplementary Material.

2.2 BPPart Algorithm

In this section, we give a dynamic programming algorithm, `BPPart`, to compute the partition function. It is well-known that the partition function can be computed by developing similar recursions as the one introduced in the simpler base-pair maximization models, such as `BPMax` and `IRIS`, with two simple modifications. The first is that algebraically, we operate with the field of reals rather than the max-plus semi-ring. Here, the additive identity is 0, rather than `INT_MIN` and the multiplicative identity is 1, rather than 0. The second is that because addition is not idempotent, we must carefully ensure that we enumerate substructures in a mutually exclusive manner. Before starting to explain the algorithm and its recursions, we have to mention that similar and equivalent (except for the weighted base-pairs feature that is being to our model to decrease the effect of ignoring entropy) algorithms can be derived from the complete models (`piRNA` and `rip`). However, we found it easier to come with decompositions and recursions from scratch and build our 8 dynamic programming tables, rather than starting with the complete models with over 90 tables, and eliminating or merging those that capture cases not required in our simplified model. This also helps us to come up with less and cleaner equations, and avoid any potential problems in reducing those methods to solve our problem. Still the overall structure of the algorithms would probably seem similar due to their common approach toward computing the partition function, namely decomposing more general structure to simpler ones and using dynamic programming.

First, we start with the recursions for computing the partition function on a single strand which is going to occur in many cases of the double-stranded version. Let $\mathrm{Q}_{i,j}$ represent the partition function of the subsequence $[i,j]$. As shown in Figure 2, there are two mutually exclusive cases: either (the right case) there is no arc, or (the left case) there is a unique leftmost arc (the cyan fill ensures this) which starts at k, and a substructure on $[k,j]$ with an arc starting at k, for which we introduce a new table Qz.

To define $\mathrm{Qz}_{i,j}$, let $i \bullet k$ (read as "let i arc k") for some index k. This results in two Q substructures, one on $[i+1,k-1]$, and the other on $[k+1,j]$. Therefore, the value of $\mathrm{Qz}_{i,j}$ can be computed using Equation (7) which accounts for the assumption that no pairing is allowed between two bases that are less than 3 bases apart:

$$\mathrm{Q}_{i,j} = \begin{cases} 1 & j \leq i \\ 1 + \displaystyle\sum_{k=i}^{j-4} \mathrm{Qz}_{k,j} & \text{otherwise,} \end{cases} \tag{6}$$

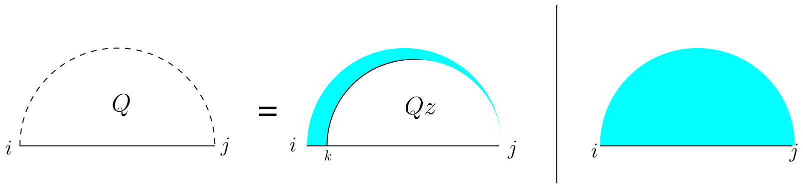

Figure 2 For computing Q, notice that either there is no pairing or there is at least one arc which starts at some index k and results in a case of Qz.

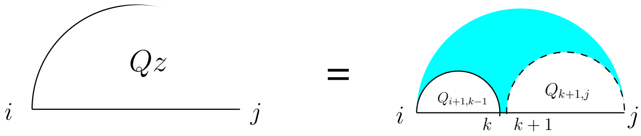

Figure 3 Computing Qz can be achieved by considering the base k that is paired with i and the two Q substructures it forms, one between i and k and one after k.

$$\mathrm{Qz}_{i,j} = \begin{cases} 0 & j - i < 4 \\ \displaystyle\sum_{k=i+4}^{j} \mathrm{Q}_{i+1,k-1} \times e^{\mathrm{score}(i,k)} \times \mathrm{Q}_{k+1,j} & \text{otherwise.} \end{cases} \tag{7}$$

For the partition function of a pair of RNA sequences, we consider a 4-dimensional table QI in which $\mathrm{QI}_{i_1,j_1,i_2,j_2}$ is the value of base pair counting partition function for the subsequences $[i_1, j_1]$ on $\mathbf{R}$ and $[i_2, j_2]$ on $\mathbf{S}$. As Figure 4 shows, we can split the set of all possible structures of QI into three mutually exclusive subsets. The leftmost case shows the structures in which there exist no bonds (the first term of Equation (8). The other two cases occur when there is at least one bond, and hence, unique leftmost events on both $\mathbf{R}$ and $\mathbf{S}$, at positions k_1 and k_2, respectively. In the second (middle) case, these leftmost events are end points of a bond, $k_1 \circ k_2$; hence, this case can be broken into: a bond-free section on the left of the bond itself, and a general case of QI on the right of the bond. The third case occurs when k_1 and k_2 are not end points of a bond. We call this structure QIa, and explain it next.

$$\begin{aligned} &\mathrm{QI}_{i_1,j_1,i_2,j_2} = \\ &\quad \mathrm{Q}^{(1)}_{i_1,j_1} \times \mathrm{Q}^{(2)}_{i_2,j_2} + \\ &\quad \sum_{k_1=i_1}^{j_1} \sum_{k_2=i_2}^{j_2} L_{i_1,j_1,k_1,i_2,j_2,k_2} + \\ &\quad \sum_{k_1=i_1}^{j_1} \sum_{k_2=i_2}^{j_2} \left(\mathrm{Q}^{(1)}_{i_1,k_1-1} \times \mathrm{Q}^{(2)}_{i_2,k_2-1} \times \mathrm{QIa}_{k_1,j_1,k_2,j_2} \right), \end{aligned} \tag{8}$$

$$L_{i_1,j_1,k_1,i_2,j_2,k_2} = \mathrm{Q}^{(1)}_{i_1,k_1-1} \times \mathrm{Q}^{(2)}_{i_2,k_2-1} \times e^{\mathrm{iscore}(k_1,k_2)} \times \mathrm{QI}_{k_1+1,j_1,k_2+1,j_2}. \tag{9}$$

For computing $\mathrm{QIa}_{i_1,j_1,i_2,j_2}$, (see Figure 5) we have to consider the property of this structure that the leftmost bases on both $\mathbf{R}$ and $\mathbf{S}$ have to be events, but they cannot both be the end points of a bond. Therefore, either one or both of them have to be end points of an interaction arc. There are two possibilities.

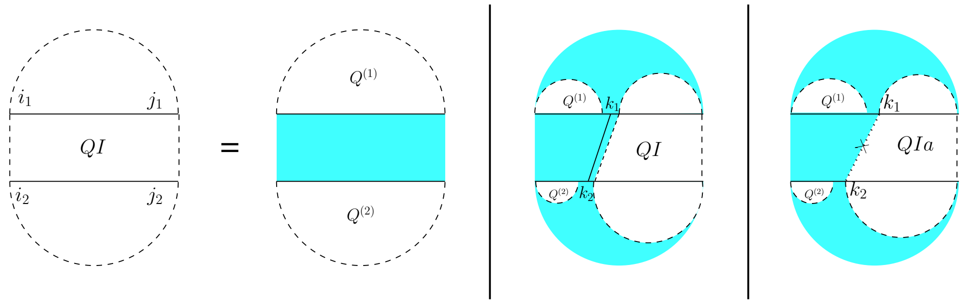

Figure 4 Each case of a QI structure (left side of the equation) can lead to three cases: either no bonds exist (leftmost case), or at least one bond exists. If the first event on both of the sequences is a bond (middle case) the subsequences to the left of the bond involve only Q and the subsequences to the right recurs on QI. Otherwise (rightmost case) we will have QIa (see Figure 5).

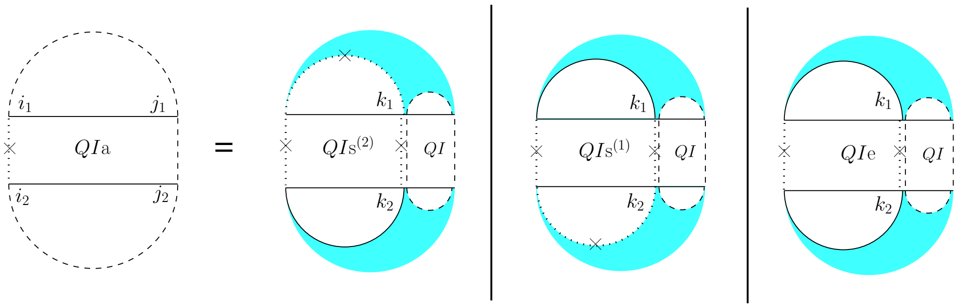

Figure 5 There are three cases for computing the QIa structure; either the leftmost base of only one of the strands is an end point of an arc or both end points are.

First, if both i_1 and i_2 are end points of some interaction arcs, $i_1 \bullet k_1$ and $i_2 \bullet k_2$, these arcs must be equivalent (or else, we have a zigzag). As shown in the rightmost diagram in Figure 5, QIa then splits into two exclusive substructures, namely one where the first and last bases on each strand are paired, and the two arcs are equivalent (we call it $\text{QIe}_{i_1,k_1,i_2,k_2}$ and derive its recursion later), followed by $\text{QI}_{k_1+1,j_1,k_2+1,j_2}$ on the suffixes of these arcs.

Otherwise, exactly one of the leftmost events on **R** and **S** is an end point of a bond, and we have two symmetric cases ($\text{QIs}^{(1)}$ and $\text{QIs}^{(2)}$), one where the interaction arc is in **R**, and the other where it is in **S**. In the first case (middle diagram in Figure 5), let k_1 be the event in **R** such that $i_1 \bullet k_1$ is an interaction arc, and $[i_2, k_2]$ is the longest subsequence in **S** that $i_1 \bullet k_1$ subsumes, and k_2 is an event. The suffix of this substructure recurs on QI. We derive $\text{QIs}^{(1)}$ later.

To derive $\text{QIe}_{i_1,j_1,i_2,j_2}$, note that removing the arcs $i_1 \bullet j_1$ and $i_2 \bullet j_2$ yields the general case of $\text{QI}_{i_1+1,j_1-1,i_2+1,j_2-1}$ for the inner-section with an additional constraint that there has be at least one bond in that region because the assumption is that the extracted arcs were interaction arcs. We can fulfill this constraint by excluding all cases where no bonds exist (i.e., considering only the two rightmost substructures of Figure 4).

To derive $QIs^{(1)}_{i_1,j_1,i_2,j_2}$ let k_1 be the leftmost event in the subsequence $[i_1 + 1, j_1 - 1]$. Note that such a k_1 is guaranteed to exist because first, $i_1 \bullet j_1$ subsumes $[i_2, j_2]$ and we know that i_2 is an event, i.e., the end point of either a bond (subsumed by $i_1 \bullet j_1$) or of an interaction arc. Then (see Figure 6) we define a new substructure, $\text{QIaux}^{(1)}$, after removing $i_1 \bullet j_1$ and the prefix of **R** up to k_1.

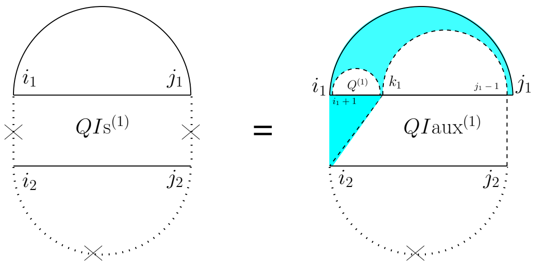

Figure 6 QIs$^{(1)}$ has one arc that can be extracted and the structure derived will have the property that the two end bases of the bottom strand cannot be paired (the new structure inherits this property from QIs$^{(1)}$). On the top strand, we consider the leftmost event. This new structure is QIaux$^{(1)}$.

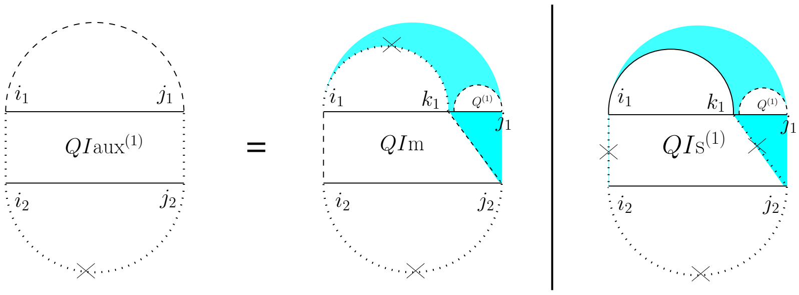

Figure 7 Two cases must be considered for the QIaux$^{(1)}$ structure, in which the two end points of the bottom strand are events. For the top strand, only the leftmost end point is required to be an event. It can either be the end point of an arc (rightmost case) or not (leftmost case).

To derive QIaux$^{(1)}_{i_1,j_1,i_2,j_2}$, note that the context of its definition implies that i_1, i_2 and j_2 are all three events. Let, as shown in Figure 7, k_1 be the *last* event on $[i_1, j_1]$. Now, if $i_1 \bullet k_1$, then recur on QIs$^{(1)}$. Otherwise, k_1 is an event that does not pair with i_1. We define a new substructure, QIm, where all four corners are events, and neither $i_1 \bullet j_1$ nor $i_2 \bullet j_2$ is allowed.

For computing QIm$_{i_1,j_1,i_2,j_2}$, since there are four corners each of which can be the end point of either a bond or of an arc, there might be at most sixteen possibilities. Upon combining some of those sixteen possibilities, we have to consider four mutually exclusive cases (see Figure 8). The first one is the case where $i_1 \circ i_2$ and $j_1 \circ j_2$ and the remaining part will be QI$_{i_1+1,j_1-1,i_2+1,j_2-1}$. That case corresponds to all four corner events being the end points of bonds. Since we assume there are no crossing bonds, we must have $i_1 \circ i_2$ and $j_1 \circ j_2$. In the second case, i_1 and i_2 are the end points of a bond, i.e., $i_1 \circ i_2$, but either j_1 or j_2 (or both) does not form a bond. That captures three out of the sixteen total possibilities. Since j_1 and j_2 are both events but do not form a bond, we define a term QIac which is the sum of QIe and the two symmetric QIs's, since they preserve the constraints that arise in the first term in the definition of QIa (see Figure 5). Note that we do not need a separate dynamic programming table for QIac because it can simply be replaced with the sum of the terms it represents. However, using this terms helps us to keep the equations easier to follow. The prefix of this substructure in the second case is a general recursion on QI on the subsequences $[i_1 + 1, k_1 - 1]$ and $[i_2 + 1, k_2 - 1]$. The third case is the symmetric case of the second case, i.e., there is no bond between i_1 and i_2, but $j_1 \circ j_2$. The prefix of this bond is a recursion on QIa. That captures three out of the sixteen total possibilities. Finally, the

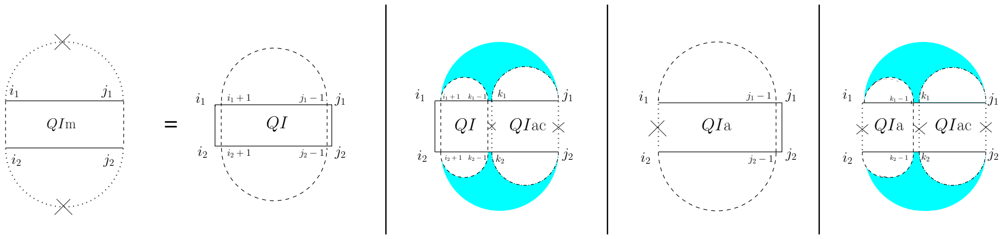

Figure 8 For computing QIm, since we know the four end points are events, but none of the two end points in one strand can form an arc, we must consider the four different cases shown above. For convenience, arcs of QIac structure are shown with dash-dotted lines because it represents the sum of three structures in which each of the arcs could be present or not (we could replace the second and fourth cases with three cases, one for each term of Equation (11)).

fourth case corresponds to either i_1 or i_2 (or both) does not form a bond and either j_1 or j_2 (or both) does not form a bond. That captures the remaining nine out of the sixteen total possibilities.

Putting all those together, we obtain

$$\mathrm{QIa}_{i_1,j_1,i_2,j_2} = \sum_{k_1=i_1}^{j_1} \sum_{k_2=i_2}^{j_2} \mathrm{QIac}_{i_1,k_1,i_2,k_2} \times \mathrm{QI}_{k_1+1,j_1,k_2+1,j_2}, \tag{10}$$

$$\mathrm{QIac}_{i_1,j_1,i_2,j_2} = \mathrm{QIs}^{(1)}_{i_1,j_1,i_2,j_2} + \mathrm{QIs}^{(2)}_{i_1,j_1,i_2,j_2} + \mathrm{QIe}_{i_1,j_1,i_2,j_2}, \tag{11}$$

$$\mathrm{QIe}_{i_1,j_1,i_2,j_2} = \begin{cases} 0 & j_1 - i_1 < 4 \\ & \text{or} \quad j_2 - i_2 < 4 \\ M_{i_1,j_1,i_2,j_2} & \text{otherwise,} \end{cases} \tag{12}$$

$$M_{i_1,j_1,i_2,j_2} = \left(\mathrm{QI}_{i_1+1,j_1-1,i_2+1,j_2-1} - \mathrm{Q}^{(1)}_{i_1+1,j_1-1} \times \mathrm{Q}^{(2)}_{i_2+1,j_2-1}\right) \times e^{\mathrm{score}(i_1,j_1)+\mathrm{score}(i_2,j_2)}, \tag{13}$$

$$\mathrm{QIs}^{(1)}_{i_1,j_1,i_2,j_2} = \begin{cases} 0 & j_1 - i_1 < 4 \quad \text{or} \quad j_2 < i_2 \\ \displaystyle\sum_{k_1=i_1+1}^{j_1-1} \mathrm{Q}^{(1)}_{i_1+1,k_1-1} \times e^{\mathrm{score}(i_1,j_1)} \times \mathrm{QIaux}^{(1)}_{k_1,j_1-1,i_2,j_2} & \text{otherwise,} \end{cases} \tag{14}$$

$$\mathrm{QIs}^{(2)}_{i_1,j_1,i_2,j_2} = \begin{cases} 0 & j_1 < i_1 \quad \text{or} \quad j_2 - i_2 < 4 \\ \displaystyle\sum_{k_2=i_2+1}^{j_2-1} \mathrm{Q}^{(2)}_{i_2+1,k_2-1} \times e^{\mathrm{score}(i_2,j_2)} \times \mathrm{QIaux}^{(2)}_{i_1,j_1,k_2,j_2-1} & \text{otherwise,} \end{cases} \tag{15}$$

$$\mathrm{QIaux}^{(1)}_{i_1,j_1,i_2,j_2} = \sum_{k_1=i_1}^{j_1} \left(\mathrm{QIs}^{(1)}_{i_1,k_1,i_2,j_2} + \mathrm{QIm}_{i_1,k_1,i_2,j_2}\right) \times \mathrm{Q}^{(1)}_{k_1+1,j_1}, \tag{16}$$

$$\mathrm{QIaux}^{(2)}_{i_1,j_1,i_2,j_2} = \sum_{k_2=i_2}^{j_2} \left(\mathrm{QIs}^{(2)}_{i_1,j_1,i_2,k_2} + \mathrm{QIm}_{i_1,j_1,i_2,k_2}\right) \times \mathrm{Q}^{(2)}_{k_2+1,j_2}, \tag{17}$$

$$\mathrm{QIm}_{i_1,j_1,i_2,j_2} = \begin{cases} e^{\mathrm{iscore}(i_1,i_2)} & i_1 = j_1 \quad \text{and} \quad i_2 = j_2 \\ N_{i_1,j_1,i_2,j_2} & i_1 < j_1 \quad \text{and} \quad i_2 < j_2 \\ 0 & \text{otherwise,} \end{cases} \tag{18}$$

$$\begin{aligned} &N_{i_1,j_1,i_2,j_2} = \\ &\quad e^{\mathrm{iscore}(i_1,i_2)+\mathrm{iscore}(j_1,j_2)} \times \mathrm{QI}_{i_1+1,j_1-1,i_2+1,j_2-1} + \\ &\quad e^{\mathrm{iscore}(i_1,i_2)} \times \sum_{k_1=i_1+1}^{j_1} \sum_{k_2=i_2+1}^{j_2} \mathrm{QI}_{i_1+1,k_1-1,i_2+1,k_2-1} \times \mathrm{QIac}_{k_1,j_1,k_2,j_2} + \\ &\quad e^{\mathrm{iscore}(j_1,j_2)} \times \mathrm{QIa}_{i_1,j_1-1,i_2,j_2-1} + \\ &\quad \sum_{k_1=i_1}^{j_1} \sum_{k_2=i_2}^{j_2} \mathrm{QIa}_{i_1,k_1,i_2,k_2} \times \mathrm{QIac}_{k_1+1,j_1,k_2+1,j_2}. \end{aligned} \tag{19}$$

3 Results

To investigate the correlation between the scores of BPPart and BPMax, and those of piRNA, we used the RISE database [16] which combines information about interacting RNAs from multiple experiments. For the human dataset, we extracted all the interaction windows for those pairs that have this data in RISE. We eliminated the ones with an interaction window size of less than 15 because they are too short for an unbiased comparison. Then, we sorted the remaining pairs based on the product of the lengths of the interacting windows (which is the base of the term that appears in the time-complexity of the algorithms). Finally, the first 50,500 pairs of sequences were chosen as our primary dataset for different experiments and analysis.

We ran piRNA on our primary dataset at $37°C$. In order to run BPPart on this dataset, we first have to choose the range of values that we want to explore for the weights of each base-pair. In general, we want to use the *stack* energies of the Turner model as a starting point for computing this range. Since the parameters form a projective space (invariant results with respect to scaling), we considered a fixed weight of 3 for CG (and GC). Using the experimentally computed stack energies of the Turner model, minimum and maximum values for the weights of AU and GU were computed. That is, to compute the maximum weight of AU (and UA), we consider the maximum released energy when AU (or UA) is stacked with another pair; this happens when UA is stacked with CG and 2.4 $kcal/mol$ energy is released. Then, we considered the minimum value of released energy in an stack for CG or GC (for which we assumed a constant weight of 3), which is 1.4 $kcal/mol$. We derived the maximum weight of AU and UA as 5.143 by multiplying 2.4 by $\frac{3}{1.4}$. Finally, we made sure that the range of values that we explore for the weight of AU and UA contains this maximum value (we chose 5.5 as the upper-bound). For finding the minimum weight of AU and UA, we consider their minimum stack energy, which is 0.6 $kcal/mol$. Given the maximum energy of CG, namely

3.4 $kcal/mol$, the value of interest is computed as $0.6 \times \frac{3}{3.4} = 0.529$. However, for the sake of comprehensiveness and exploring the shape of the plots, we used much smaller lower-bound, -4.5, for our explorations.

Assuming a fixed weight of 3 for `CG`, we computed the Pearson and Spearman's Rank correlations with the scores from `piRNA`, for all the combinations of weights of `AU` and `GU` in steps of 0.5. When computing the correlations, to normalize the scores from all algorithms, we divide them by the sum of the lengths of corresponding sequences, $L_R + L_S$. This normalization mitigates the effect of length bias on the computed correlations. This step is necessary because, generally, as the length of the pair of sequences increases the scores of all three algorithms increases, and if unnormalized scores are used, a biased higher correlation will be derived. Note that for partition functions, `piRNA` and `BPPart`, we are computing the *log* of the scores; that is why we factor out the sum of the lengths for normalization. If the original values were to be used, we would have to take the $(L_R + L_S)$th roots of the scores. Figure 9 (a) shows the Pearson correlations for different combinations of weights of `AU` and `GU` at $37°C$. Figure 9 (b) shows the scatter plot of the scores for the best combination of weights, which are 0.5, 1.0, and 3 for `AU`, `GU`, and `CG`, respectively. In this plot, the red line shows the regression line that is fitted to the points by minimizing the mean squared error (MSE). We performed the same steps and analysis for `BPMax` method (more details on this method can be found in the Supplementary Material). The optimum values of correlation are presented in Table 1. As the results show, there is a high correlation between `piRNA` and `BPPart` as well as between `piRNA` and `BPMax`.

Table 1 Correlations between `piRNA` and `BPPart` and between `piRNA` and `BPMax` at $37°C$.

Method	Pearson	Spearman's Rank
BPPart	0.855	0.864
BPMax	0.836	0.830

To make sure that the base-pair weights derived by our optimization approach are not data-dependent, in spite of the our observation of very similar optimization plots on smaller portions of the primary data, we conducted the same experiments for randomly generated sequences. To factor out the effect of length, for each pair in our primary dataset, we generated a pair of random sequences with the same lengths as those of the pair in our primary dataset. Our results show similar optimized weights, but lower correlations on this dataset (this will be discussed in the next section). More details on the results for this dataset are provided in the Supplementary Material.

To better understand the behavior of the surface around the higher values in the correlation plot of Figure 9 (a) and Figure 15 (b) in the Supplementary Material, we computed the Shannon entropy for the values above a threshold. Figure 10 shows these values for the top 30 values of Pearson and Spearman's Rank correlation at each temperature. We discuss these results in the the next section.

Finally, we designed a pipeline for generating hypothesis about the roles of RNAs in different diseases using our newly developed algorithm which makes large-scale analysis of RRI datasets practical in a reasonable time (3 hours vs. one month using `piRNA`) with reasonable resources (6.6 GB of RAM vs. about 70 GB of RAM for `piRNA`). We elaborate on this pipeline and our results in the Supplementary Material.

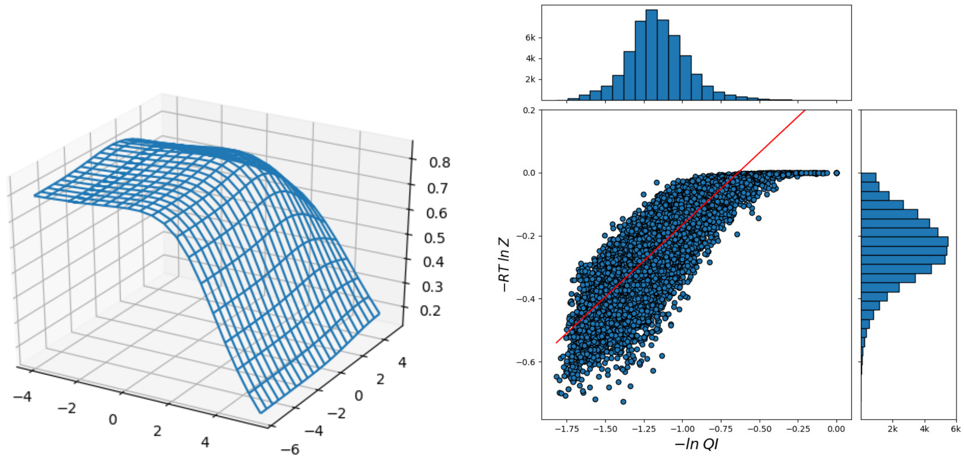

Figure 9 (a) Pearson correlation between `piRNA` and `BPPart` (vertical axis), on the primary dataset, at 37°C for different weights of AU (left axis) and GU (right axis). The weight of CG pair is fixed to 3. (b) Scatter plot of the scores form `piRNA` (y-axis) and `BPPart` (x-axis) at 37°C. The read line is fitted to the points to minimize the Mean Squared Error (MSE).

4 Discussion

Note that we can rewrite equation 3 as the following

$$-\frac{\Delta G}{T} = -\frac{\Delta H}{T} + \Delta S, \tag{20}$$

and it is clear that as $T \to 0$, $-\Delta H/T \to \infty$ and the contribution of ΔS is diminished to 0 since it is finite. Hence, at low temperatures, the effect of entropy becomes negligible, and we expect to see strong correlation between the base pair counting model and full thermodynamic model. To verify that the scores computed with our models follow this theoretical observation, we computed the correlations at different temperatures, ranging from -180 (°C) to 37 (°C) (at temperatures lower than -180 (°C) the implementation of `piRNA` was unstable and resulted in NaN values, which prevented us from computing the correlation values). Figure 11 shows the Pearson correlations between `BPPart` and `BPMax` scores with `piRNA` scores for for their best combination of base-pair weights at 37 (°C). These optimum weights for `BPPart` are 0.5, 1.0, and 3 for `AU`, `GU`, and `CG`, respectively, and for `BPMax` are 1.0, 1.5, and 3 for `AU`, `GU`, and `CG`, respectively.

Perfectly conforming with the theory, we see higher correlations at low temperatures. These results, also, somewhat validates our implementations as `piRNA` was written totally independently more than 10 years ago. Moreover, by comparing Figure 9 (a) to Figure 12, and Figure 15 (b) to Figure 15 (a), we notice that the surface around the optimum value for higher temperatures becomes flatter. Figure 10, which shows the entropy of the top 30 correlation values, confirms this observation; this means the correlation values are less sensitive to a change in the weights of the base pairs as the temperature increases; this conforms with the theory because at higher temperatures, the thermodynamic entropy increases and the total score of `piRNA` becomes less sensitive to the energy released by pairings. This means that slight changes to our optimum weights at the body temperature, are less susceptible to result in different correlations than the optimum possible correlations that can be achieved by using the optimum weights.

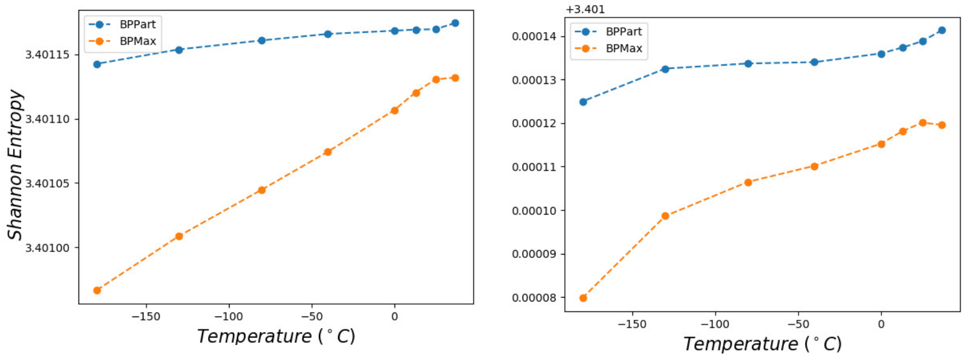

Figure 10 Shannon entropy for the top 30 Pearson (left) and Spearman's rank (right) correlation values at different temperatures for `BPPart` and `BPMax`.

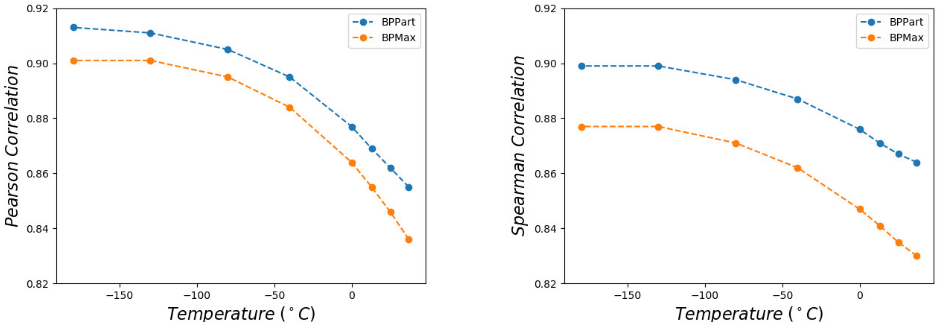

Figure 11 Pearson correlation (left) and Spearman's rank correlation (right) between `piRNA` and `BPPart` and between `piRNA` and `BPMax` at different temperatures.

Another noticeable characteristic of the optimization plots in Figures 9 (a) and 15 is the region in which the scores of both `AU` and `GU` are non-positive. This region for `BPMax` is flat because when both of these pairs are penalized (or not rewarded when their score is zero), the algorithm simply avoids making such pairs because it is trying to maximize the score. Therefore, it only tries to maximize the number of `CG` pairs, which is independent of the scores (penalty in this case) of the other two types of base pairs. This also applies to the case where one of the base pairs has a non-positive score; in that case, `BPMax` works independently of the score of that base pair. So, as soon as any of the scores becomes non-positive, `BPMax` remains constant along the corresponding axis. For `BPPart`, however, the story is different because it simply counts all the possible pairings and even if the score of a base pair becomes negative, it does not ignore counting that.

Moreover, `BPPart` has a higher correlation than `BPMax` does, which comes with the price of a 6× increase in empirical running time. Also, as Figure 10 shows, the Shannon entropy for the top 30 values is less in `BPMax` and the gap between them grows as temperature decreases; this shows that `BPPart` has a flatter region around the optimum value and its optimum correlation is less sensitive to changes in the optimum weights. Hence, we now have three choices in increasing order of computational cost: `BPMax`, `BPPart`, and `piRNA`. The computation time increases by about 6× and 225×, respectively, from one to the next on the primary dataset. We also compared their costs on a single pair of sequences, each with a length of 100 bases. It took about 1, 6, and 1200 minuets and about 0.2 GB, 1.8 GB, and 18.5 GB of RAM for BPMax, BPPart, and piRNA, respectively, to compute the score of interaction. Note that here BPPart was about 200× faster than piRNA because the sequences had equal lengths, and the terms of degree four in the length of one of the sequences that appear in the time-complexity of piRNA (mentioned in the first section) do not make a difference here.

Given the higher correlations and less sensitivity to the optimum base-pair weights, paying the extra cost (compared to BPMax) to use BPPart seems justifiable in many applications. Another important benefit of partition functions, such as BPPart and piRNA over base-pair maximization models (e.g., BPMax) is that they can be used to compute probabilities that a base is paired or remains unpaired since we have the total counts for both cases; this property becomes necessary when working with tools such as rip [21] and biRNA [7]. Moreover, when studying the effects of SNPs and variants (e.g., the pipeline we have included in the Supplementary Material) on RNA-RNA interaction, BPMax cannot replace partition functions that are more sensitive to small perturbations.

Finally, based on the results of the experiments on both the primary dataset and the random one, we see that although the shapes of the optimization plots and the optimum weights are very similar, the correlation values are less for the random dataset. This observation is probably due to the fact that interaction regions are more complementary than the random sequences of the same size. When the genomic sequences are more complementary, the effect of the energy released by pairing becomes more significant than the energy added by an increase in the entropy on the final score of piRNA. In randomly generated sequences, however, BPPart and BPMax do not capture the increase in the entropy that leads to higher energy, which makes the interaction less desirable. With this effect, BPPart and BPMax, might overestimate the score of interaction among two non-interacting regions. It is worth mentioning that using the weighted base-pairs has diminished this effect because they are optimized to generate more similar scores to the ones from complete models that consider entropy. This hypothesis has to be thoroughly investigated in the future.

5 Conclusion

We revisited the problems of partition function and structure prediction for interacting RNAs. We simplified the energy model by ignoring the effects of entropy and reduced the full-thermodynamic model into a simple weighted base-pair counting one to obtain BPPart for the partition function. As a result, BPPart runs about 225× faster than piRNA does. Hence, we gained significant speedup by potentially sacrificing accuracy. To evaluate practical accuracy of our new model, we computed the Pearson and Spearman's Rank correlations between the results of BPPart and piRNA on 50,500 experimentally characterized RRIs in the RISE database [16]. Results highly correlate with those of piRNA. At the room and body temperatures, there is considerable correlation and therefore, significant information in the results of BPPart.

We conclude that our simpler models captures a significant portion of the thermodynamic information. Its considerable speedup and simplicity enables its use-cases in larger-scale studies which were not feasible with comprehensive models in reasonable time and resources. This approach for simplifying the full thermodynamic models can also be used together with other approximation methods that are based on thermodynamic models. Also, the information captured by BPPart can possibly be used to introduce physics-guided information that may complement more complex prediction models in the future. We introduced a pipeline which becomes practical with our faster model and might be useful to explain how some mutations lead to some specific phenotypic consequences.

References

1 Can Alkan, Emre Karakoc, Joseph H Nadeau, S Cenk Sahinalp, and Kaizhong Zhang. Rna–rna interaction prediction and antisense rna target search. *Journal of Computational Biology*, 13(2):267–282, 2006.

2 Mirela Andronescu, Zhi Chuan Zhang, and Anne Condon. Secondary structure prediction of interacting rna molecules. *Journal of molecular biology*, 345(5):987–1001, 2005.

3 Stephan H Bernhart, Hakim Tafer, Ulrike Mückstein, Christoph Flamm, Peter F Stadler, and Ivo L Hofacker. Partition function and base pairing probabilities of rna heterodimers. *Algorithms for Molecular Biology*, 1(1):1–10, 2006.

4 Doron Betel, Anjali Koppal, Phaedra Agius, Chris Sander, and Christina Leslie. Comprehensive modeling of microrna targets predicts functional non-conserved and non-canonical sites. *Genome biology*, 11(8):R90, 2010.

5 Anke Busch, Andreas S Richter, and Rolf Backofen. Intarna: efficient prediction of bacterial srna targets incorporating target site accessibility and seed regions. *Bioinformatics*, 24(24):2849–2856, 2008.

6 Song Cao and Shi-Jie Chen. Predicting rna pseudoknot folding thermodynamics. *Nucleic acids research*, 34(9):2634–2652, 2006.

7 Hamidreza Chitsaz, Rolf Backofen, and S Cenk Sahinalp. birna: Fast rna-rna binding sites prediction. In *International Workshop on Algorithms in Bioinformatics*, pages 25–36. Springer, 2009.

8 Hamidreza Chitsaz, Raheleh Salari, S.Cenk Sahinalp, and Rolf Backofen. A partition function algorithm for interacting nucleic acid strands. *Bioinformatics*, 25(12):i365–i373, 2009. Also ISMB/ECCB proceedings.

9 Ilaria Di Donato, Silvia Bianchi, Nicola De Stefano, Martin Dichgans, Maria Teresa Dotti, Marco Duering, Eric Jouvent, Amos D Korczyn, Saskia AJ Lesnik-Oberstein, Alessandro Malandrini, et al. Cerebral Autosomal Dominant Arteriopathy with Subcortical Infarcts and Leukoencephalopathy (CADASIL) as a model of small vessel disease: update on clinical, diagnostic, and management aspects. *BMC medicine*, 15(1):41, 2017.

10 Laura DiChiacchio, Michael F Sloma, and David H Mathews. Accessfold: predicting rna–rna interactions with consideration for competing self-structure. *Bioinformatics*, 32(7):1033–1039, 2015.

11 Roumen A Dimitrov and Michael Zuker. Prediction of hybridization and melting for double-stranded nucleic acids. *Biophysical Journal*, 87(1):215–226, 2004.

12 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.

13 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.

14 Ali Ebrahimpour-Boroojeny, Sanjay Rajopadhye, and Hamidreza Chitsaz. Bppart and bpmax: Rna-rna interaction partition function and structure prediction for the base pair counting model. *arXiv preprint*, 2019. `arXiv:1904.01235`.

15 Dimos Gaidatzis, Erik van Nimwegen, Jean Hausser, and Mihaela Zavolan. Inference of mirna targets using evolutionary conservation and pathway analysis. *BMC bioinformatics*, 8(1):69, 2007.

16 Jing Gong, Di Shao, Kui Xu, Zhipeng Lu, Zhi John Lu, Yucheng T Yang, and Qiangfeng Cliff Zhang. Rise: a database of rna interactome from sequencing experiments. *Nucleic acids research*, 46(D1):D194–D201, 2018.

17 Mitchell Guttman, Ido Amit, Manuel Garber, Courtney French, Michael F Lin, David Feldser, Maite Huarte, Or Zuk, Bryce W Carey, John P Cassady, et al. Chromatin signature reveals over a thousand highly conserved large non-coding RNAs in mammals. *Nature*, 458(7235):223–227, 2009.

18 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(2):167–188, 1994.

19 Justin Bo-Kai Hsu, Chih-Min Chiu, Sheng-Da Hsu, Wei-Yun Huang, Chia-Hung Chien, Tzong-Yi Lee, and Hsien-Da Huang. mirtar: an integrated system for identifying mirna-target interactions in human. *BMC bioinformatics*, 12(1):300, 2011.

20 Hong Ming Hu, Karen O'Rourke, Mark S Boguski, and Vishua M Dixit. A novel RING finger protein interacts with the cytoplasmic domain of CD40. *Journal of Biological Chemistry*, 269(48):30069–30072, 1994.

21 Fenix WD Huang, Jing Qin, Christian M Reidys, and Peter F Stadler. Partition function and base pairing probabilities for rna–rna interaction prediction. *Bioinformatics*, 25(20):2646–2654, 2009.

22 Anne Joutel, Christophe Corpechot, Anne Ducros, Katayoun Vahedi, Hugues Chabriat, Philippe Mouton, Sonia Alamowitch, Valérie Domenga, Michaelle Cecillion, Emmanuelle Marechal, et al. Notch3 mutations in cerebral autosomal dominant arteriopathy with subcortical infarcts and leukoencephalopathy (CADASIL), a mendelian condition causing stroke and vascular dementia. *Annals of the New York Academy of Sciences*, 826(1):213–217, 1997.

23 Yuki Kato, Tatsuya Akutsu, and Hiroyuki Seki. A grammatical approach to rna–rna interaction prediction. *Pattern Recognition*, 42(4):531–538, 2009.

24 Stephanie Kehr, Sebastian Bartschat, Peter F Stadler, and Hakim Tafer. Plexy: efficient target prediction for box c/d snornas. *Bioinformatics*, 27(2):279–280, 2010.

25 Michael Kertesz, Nicola Iovino, Ulrich Unnerstall, Ulrike Gaul, and Eran Segal. The role of site accessibility in microrna target recognition. *Nature genetics*, 39(10):1278, 2007.

26 Azra Krek, Dominic Grün, Matthew N Poy, Rachel Wolf, Lauren Rosenberg, Eric J Epstein, Philip MacMenamin, Isabelle Da Piedade, Kristin C Gunsalus, Markus Stoffel, et al. Combinatorial microrna target predictions. *Nature genetics*, 37(5):495, 2005.

27 Jan Krüger and Marc Rehmsmeier. Rnahybrid: microrna target prediction easy, fast and flexible. *Nucleic acids research*, 34(suppl_2):W451–W454, 2006.

28 Almin I Lalani, Carissa R Moore, Chang Luo, Benjamin Z Kreider, Yan Liu, Herbert C Morse, and Ping Xie. Myeloid cell TRAF3 regulates immune responses and inhibits inflammation and tumor development in mice. *The Journal of Immunology*, 194(1):334–348, 2015.

29 Ronny Lorenz, Stephan H Bernhart, Christian Hoener Zu Siederdissen, Hakim Tafer, Christoph Flamm, Peter F Stadler, and Ivo L Hofacker. Viennarna package 2.0. *Algorithms for Molecular Biology*, 6(1):26, 2011.

30 Martin Mann, Patrick R Wright, and Rolf Backofen. Intarna 2.0: enhanced and customizable prediction of rna–rna interactions. *Nucleic acids research*, 45(W1):W435–W439, 2017.

31 NR Markham, M Zuker, and JM Keith. Unafold: software for nucleic acid folding and hybridization., pp. 3–31, 2008.

32 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.

33 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.

34 Kevin C Miranda, Tien Huynh, Yvonne Tay, Yen-Sin Ang, Wai-Leong Tam, Andrew M Thomson, Bing Lim, and Isidore Rigoutsos. A pattern-based method for the identification of microrna binding sites and their corresponding heteroduplexes. *Cell*, 126(6):1203–1217, 2006.

35 Ulrike Mückstein, Hakim Tafer, Jörg Hackermüller, Stephan H Bernhart, Peter F Stadler, and Ivo L Hofacker. Thermodynamics of rna–rna binding. *Bioinformatics*, 22(10):1177–1182, 2006.

36 Jin-Wu Nam, Olivia S Rissland, David Koppstein, Cei Abreu-Goodger, Calvin H Jan, Vikram Agarwal, Muhammed A Yildirim, Antony Rodriguez, and David P Bartel. Global analyses of the effect of different cellular contexts on microrna targeting. *Molecular cell*, 53(6):1031–1043, 2014.

37 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.

38 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.

39 Dmitri D Pervouchine. Iris: intermolecular rna interaction search. *Genome Informatics*, 15(2):92–101, 2004.

40 Dmitri D Pervouchine. Iris: intermolecular rna interaction search. *Genome Informatics*, 15(2):92–101, 2004.

41 Martin Reczko, Manolis Maragkakis, Panagiotis Alexiou, Ivo Grosse, and Artemis G Hatzigeorgiou. Functional microrna targets in protein coding sequences. *Bioinformatics*, 28(6):771–776, 2012.

42 Marc Rehmsmeier, Peter Steffen, Matthias Höchsmann, and Robert Giegerich. Fast and effective prediction of microrna/target duplexes. *Rna*, 10(10):1507–1517, 2004.

43 Ángela Riffo-Campos, Ismael Riquelme, and Priscilla Brebi-Mieville. Tools for sequence-based mirna target prediction: what to choose? *International journal of molecular sciences*, 17(12):1987, 2016.

44 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.

45 Hakim Tafer, Stephanie Kehr, Jana Hertel, Ivo L Hofacker, and Peter F Stadler. Rnasnoop: efficient target prediction for h/aca snornas. *Bioinformatics*, 26(5):610–616, 2010.

46 Brian Tjaden. Targetrna: a tool for predicting targets of small rna action in bacteria. *Nucleic acids research*, 36(suppl_2):W109–W113, 2008.

47 Shoji Tsuji, Prabhakara V Choudary, Brian M Martin, Suzanne Winfield, John A Barranger, and Edward I Ginns. Nucleotide sequence of cdna containing the complete coding sequence for human lysosomal glucocerebrosidase. *Journal of Biological Chemistry*, 261(1):50–53, 1986.

48 Sinan Uğur Umu and Paul P Gardner. A comprehensive benchmark of rna–rna interaction prediction tools for all domains of life. *Bioinformatics*, 33(7):988–996, 2017.

49 S Patrick Walton, Gregory N Stephanopoulos, Martin L Yarmush, and Charles M Roth. Thermodynamic and kinetic characterization of antisense oligodeoxynucleotide binding to a structured mrna. *Biophysical journal*, 82(1):366–377, 2002.

50 Michael S Waterman and Temple F Smith. Rna secondary structure: A complete mathematical analysis. *Mathematical Biosciences*, 42(3-4):257–266, 1978.

51 Anne Wenzel, Erdinç Akbaşli, and Jan Gorodkin. Risearch: fast rna–rna interaction search using a simplified nearest-neighbor energy model. *Bioinformatics*, 28(21):2738–2746, 2012.

52 Wenlong Xu, Anthony San Lucas, Zixing Wang, and Yin Liu. Identifying microrna targets in different gene regions. *BMC bioinformatics*, 15(7):S4, 2014.

53 Yuanji Zhang. miru: an automated plant mirna target prediction server. *Nucleic acids research*, 33(suppl_2):W701–W704, 2005.

54 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.

A Rivas-Eddy Diagrams

Here we describe the "Rivas-Eddy diagram" notation that we adopt in this paper. The main elements are:

1. A solid horizontal straight line represents a sequence; we have two sequences drawn as two parallel horizontal lines.
2. A solid curved line between two points in the same sequence is an arc; all arcs are either above the upper sequence, or below the lower one.
3. A dotted curved line with a cross in the middle, between two points in the same sequence means that those two points *do not* form an arc.
4. A dashed curved line between two points in the same sequence denotes either 2 or 3.
5. A solid line between two points in different sequences is a bond.
6. Similarly, a dotted line with a cross in the middle, between two points in different sequences means that those two points *do not* form a bond.
7. A dashed line between two points in different sequences denotes either 5 or 6.
8. A region is the space under an arc, or between bonds. When there are no additional choices of bonds/arcs in a given region, we fill it with a color (cyan); no arc or bond crosses a filled region.
9. A point in a sequence may be labeled with an index, and in general, the set of such indices are free variables used in the recursions; the index of unlabeled points before (after) labeled points is assumed to be the predecessor (successor) of the label.
10. A diagram may be labeled with the name(s) of the constituent substructures (which are eventually implemented as dynamic programming tables/variables).
11. A vanishing arc (i.e., one that starts at some index, and does not explicitly specify an end point) represents a structure whose start point is as specified, and the end point is to be determined.

B Other Results for BPPart

For the sake of comparison of how the plots in Figure 9 would look like at $-180°C$, we generated the same plots and presented them in Figure 12.

As mentioned in the paper, we performed the same optimization procedure on randomly-generated data. Figure 13 shows these optimization plots. Notice that we shapes of the plots and optimum weights are very similar, but the correlations are less. The potential reasons for this observation are disucussed in the paper.

C BPMax Algorithm

Here, we give the dynamic programming algorithm for the `BPMax` model. When explaining some of the equations, helper functions, called H, L, M, N, are used to ease the reading of the paper. To differentiate these helper functions, superscripts are used.

For a single strand of nucleotides, we define $\mathrm{S}_{i,j}$ as the maximum weighted sum of base pair scores on all possible foldings of subsequence $[i, j]$. We need to make such a table, for each of the $\mathbf{R}$ and $\mathbf{S}$ strands, and we distinguish between them by superscripts (1) and (2), respectively. We also define F_{i_1,j_1,i_2,j_2} as the maximum weighted sum of base pair scores (both intra- and inter-pairings) of subsequences $[i_1, j_1]$ from $\mathbf{R}$ and $[i_2, j_2]$ from $\mathbf{S}$.

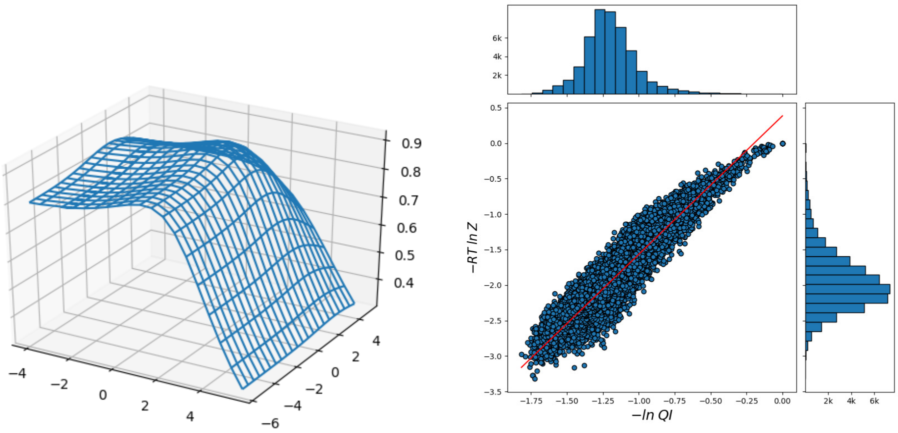

Figure 12 (a) Pearson correlation between `piRNA` and `BPPart` (vertical axis), on the primary dataset, at $-180°C$ for different weights of AU (left axis) and GU (right axis). The weight of CG pair is fixed to 3. (b) Scatter plot of the scores form `piRNA` (y-axis) and `BPPart` (x-axis) at $-180°C$. The read line is fitted to the points to minimize the Mean Squared Error (MSE).

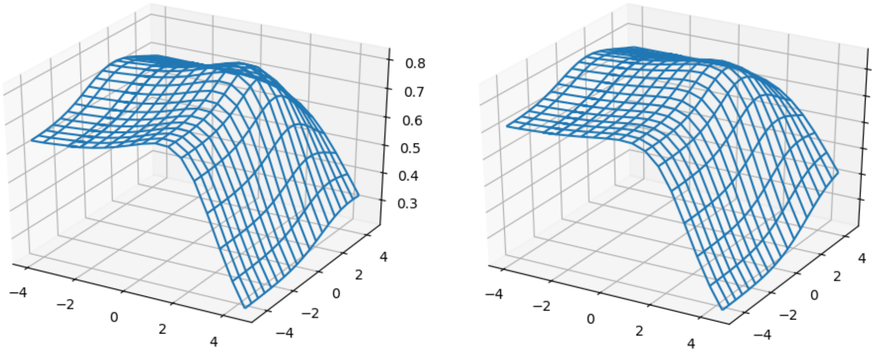

Figure 13 Pearson correlation between `piRNA` and `BPPart` (vertical axis), on the randomly generated dataset, at $-180°C$ (left) and $37°C$ (right) for different values of constant factors (weights) for AU (left axis) and GU (right axis). The weight of CG pair is fixed at 3.

The computation of $\mathrm{S}_{i,j}$ is based on the well known single RNA folding algorithm [37]. For short sequences (i.e., those whose length is strictly less than 5) the score is 0, otherwise, we use the recursion in the second case of Equation (21) shown below. It considers the case where we have an arc $i \bullet j$ and recurs on $[i+1, j-1]$, and also other cases in which the i^{th} and j^{th} bases are not paired and the $[i, j]$ is split into two smaller subsequences:

$$\mathrm{S}_{i,j} = \begin{cases} 0 & j - i < 4 \\ \max\left(\mathrm{S}_{i+1,j-1} + \mathrm{score}(i,j),\ \max_{k=i}^{j-1} \mathrm{S}_{i,k} + \mathrm{S}_{k+1,j}\right) & \text{otherwise.} \end{cases} \tag{21}$$

We define the recurrences for $\mathrm{F}_{i_1,j_1,i_2,j_2}$ similarly. When either sequence is empty, the value is simply the S of the other sequence, and for two singleton sequences, it is the score of the single bond possible. Otherwise we have three cases: (i) i_1 arcs j_1 ($i_1 \bullet j_1$) in which case the residual structure is given by a recursion on $\mathrm{F}_{i_1+1,j_1-1,i_2,j_2}$, (ii) the symmetric case of $i_2 \bullet j_2$ and $\mathrm{F}_{i_1,j_1,i_2+1,j_2-1}$, or (iii) none of these arcs, and two recursive cases of $\mathrm{F}_{i_1,k_1,i_2,k_2}$ and $\mathrm{F}_{k_1+1,j_1,k_2+1,j_2}$. They are illustrated in Figure 14, which lead to

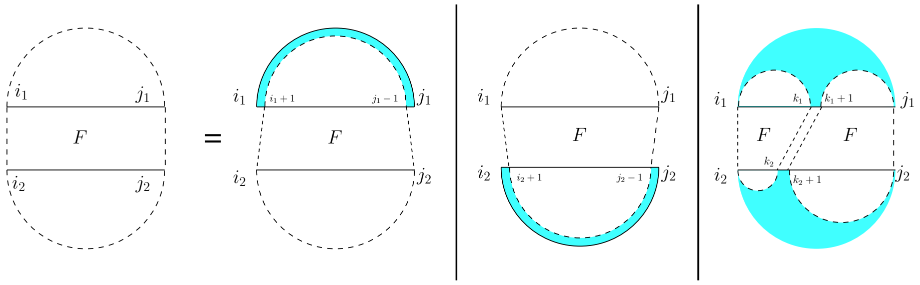

Figure 14 The four cases defining table F. Note that in the `BPMax` algorithm, the cases do not have to be mutually exclusive since we are working with the max operator, which is idempotent.

$$\mathrm{F}_{i_1,j_1,i_2,j_2} = \begin{cases} -\infty & j_1 < i_1 \text{ and } j_2 < i_2 \\ \mathrm{S}^{(1)}_{i_1,j_1} & i_1 \leq j_1 \text{ and } j_2 < i_2 \\ \mathrm{S}^{(2)}_{i_2,j_2} & j_1 < i_1 \text{ and } i_2 \leq j_2 \\ \mathrm{iscore}(i_1,i_2) & i_1 = j_1 \text{ and } i_2 = j_2 \\ \max\left[\, \mathrm{F}_{i_1+1,j_1-1,i_2,j_2} + \mathrm{score}(i_1,j_1), \right. & \\ \qquad \mathrm{F}_{i_1,j_1,i_2+1,j_2-1} + \mathrm{score}(i_2,j_2), & \\ \qquad \left. H_{i_1,j_1,i_2,j_2} \,\right] & \text{otherwise,} \end{cases} \tag{22}$$

$$H_{i_1,j_1,i_2,j_2} = \max_{k_1=i_1-1}^{j_1} \max_{k_2=i_2-1}^{j_2} (\mathrm{F}_{i_1,k_1,i_2,k_2} + \mathrm{F}_{k_1+1,j_1,k_2+1,j_2}). \tag{23}$$

Note that H is equivalent to

$$H_{i_1,j_1,i_2,j_2} = \max \left(\begin{array}{l} \mathrm{S}^{(1)}(i_1,j_1) + \mathrm{S}^{(2)}(i_2,j_2), \\ \max\limits_{k_1=i_1}^{j_1-1} \max\limits_{k_2=i_2}^{j_2-1} \mathrm{F}_{i_1,k_1,i_2,k_2} + \mathrm{F}_{k_1+1,j_1,k_2+1,j_2}, \\ \max\limits_{k_2=i_2}^{j_2-1} \mathrm{S}^{(2)}(i_2,k_2) + \mathrm{F}_{i_1,j_1,k_2+1,j_2}, \\ \max\limits_{k_2=i_2}^{j_2-1} \mathrm{F}_{i_1,j_1,i_2,k_2} + \mathrm{S}^{(2)}(k_2+1,j_2), \\ \max\limits_{k_1=i_1}^{j_1-1} \mathrm{S}^{(1)}(i_1,k_1) + \mathrm{F}_{k_1+1,j_1,i_2,j_2}, \\ \max\limits_{k_1=i_1}^{j_1-1} \mathrm{F}_{i_1,k_1,i_2,j_2} + \mathrm{S}^{(1)}(k_1+1,j_1) \end{array} \right). \tag{24}$$

In Equation (22), we compute S tables separately for each strand, according to Equation (21) with the corresponding sequence as the input, and we distinguish them by superscripts $^{(1)}$ and $^{(2)}$ above. We use the same superscript convention throughout this paper.

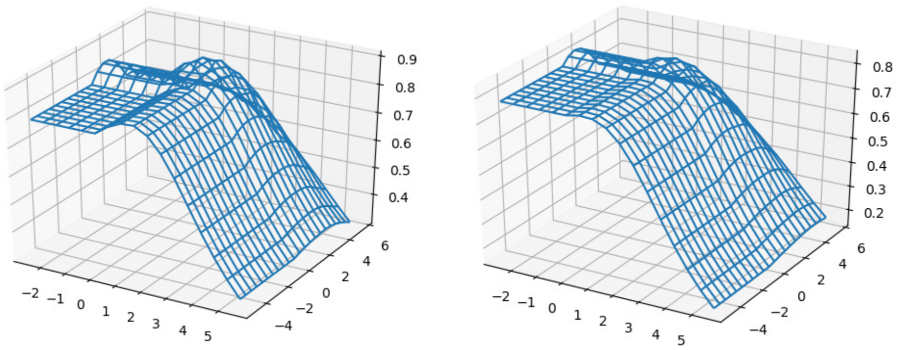

Figure 15 Pearson correlation between `piRNA` and `BPMax` (vertical axis), on the primary dataset, at $-180°C$ (left) and $37°C$ (right) for different values of constant factors (weights) for AU (left axis) and GU (right axis). The weight of CG pair is fixed at 3.

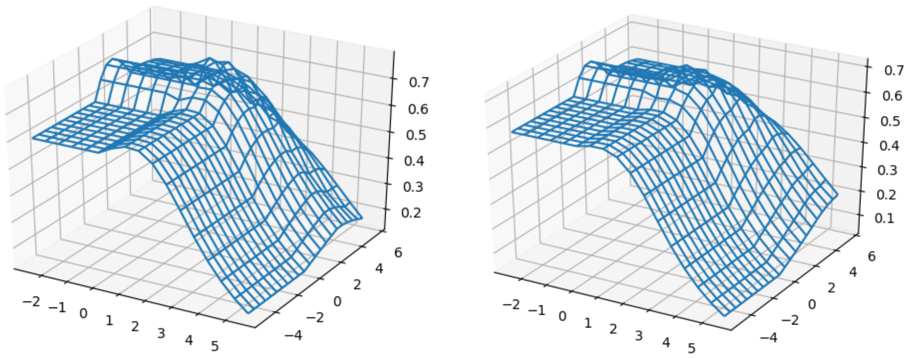

Figure 16 Pearson correlation between `piRNA` and `BPMax` (vertical axis), on the randomly generated dataset, at $-180°C$ (left) and $37°C$ (right) for different values of constant factors (weights) for AU (left axis) and GU (right axis). The weight of CG pair is fixed at 3.

C.1 Results for BPMax

The `BPMax` algorithm was about 1300× faster than *piRNA* on our primary dataset. We performed similar optimization procedure as the one explained for `BPPart` to obtain optimum weights for the base-pairs that maximize the correlations with `piRNA` scores. Figure 15 showes these optimization plots at $-180°C$ and $37°C$. We did the same analysis on randomly-generate data and presented the optimization plots in Figure 16.

D Application of BPPart in Human Biology

One of the use-cases of `BPPart` and `BPMax`, among others, is making predictions about the consequences of a slight change in the RNA sequences. This information becomes helpful for various domains and tasks, such as synthetic biology and studying the mutations. Between `BPMax` and `BPPart`, the latter is much more sensitive to small changes in the sequence, because it considers all possible structures that the two interacting sequences might form. Therefore, even a missense mutation might make a tangible difference in the computed `BPPart` score.

To verify this hypothesis, we used BPPart to study the effects of known missense mutations, provided by Ensembl, in the interaction regions of some RISE pairs. Given a pair of interacting RNAs in RISE for which the information about the interacting regions is provided, we retrieved the data of all the reported missense mutations of those regions from Ensembl API. Also, we got the phenotypic consequences of each mutation from Ensembl. Finally, we computed the BPPart score for the original sequence of one of the interacting regions and each of the mutated versions of the other sequence. Among all the generated scores for a pair, we found the outliers using the interquartile range. These outliers represent a mutation in the interacting window of one of the RNA pairs that causes a great difference in the interaction score. In the rest of this section, we almost-randomly pick and narrate two of such cases that we observed, among many discovered ones. In the arxiv version of this paper [14], we report 65 such pairs that have been discovered using this pipeline after analyzing more than one million pairs of sequences that have been generated after applying the known missense mutations to over $15,200$ pairs of interacting sequences reported in the RISE database. Further study of each of these pairs and more comprehensive study of effect of nonsense mutations on RRI would be a next step in the future.

D.1 Traces of TRAF3 in CADASIL

CADASIL is an inherited condition in which the muscle cells of small blood vessels, especially the ones in the brain, gradually die and cause many impairements, such as stroke, cognitive impairement, and mood disorders in the elderly [9]. It has been shown that mutation in NOTCH3, which resides on the reverse strand of chromosome 19, is responsible for this condition in people with this genetic disorder [22]. NOTCH3 and TRAF3 are a pair of interacting RNAs that have been reported in RISE. One of the missense mutations in NOTCH3 that has been reported to be contributing to CADASIL [22] lies within the interacting region of this gene, from loci $15,161,520$ to $15,161,543$ (according to GRCh38 assembly of human genome), with TRAF3. Interestingly, this mutation, which replaces nucleotide C with G at loci $15,161,526$ of chromosome 19, causes a dramatic increase in the score of BPPart such that it makes it an outlier when the aformentioned procedure is followed. TRAF3 is a gene that has been reported to play a role in angiogenesis [20, 28]. A noticeable increase in the score of BPPart increases the chance that these two RNAs interact and cause post-transcriptional conditions that affect the translation rate of TRAF3 which possibly contributes to the phenotypic consequences of CADASIL. Further evaluation and verification of this hypothesis requires further experimental analysis.

D.2 Traces of SNORD3D in Parkinson’s Disease

SNORD3D is a small nucleolar RNA which has been detected not long ago [17] with which no specific task or annotation is associated in the literature yet. According to the RISE database, one of the genes that interacts with this snoRNA is GBA. Mutations in GBA has been reported to play a role in Parkinson’s disease which is a brain disorder that affects movement and often causes tremors. One of the GBA mutations that is reported to be linked with Parkinson’s disease lies within the interaction region of this gene, from loci $155,239,966$ to $155,239,984$ (according to GRCh38 assembly of human genome), with SNORD3D. This specific mutation of GBA, which changes the nucleotide G to C at loci $155,239,972$ of chromosome 1, is one of the cases that is detected as an outlier using our aforementioned analysis of BPPart scores. This mutation, when applied to GBA, decreases its score of interaction with SNORD3D, which might cause the interaction to occur much less than the normal case. This possibly leads to a change in the expression of GBA protein. According to KEGG, GBA is a member of Other glycan degradation, Sphingolipid metabolism, Metabolic pathways, and Lysosome pathways [47]. Therefore, we hypothesize the role of SNORD3D in some or all of those pathways, particularly, the ones that are closely related to Parkinson's disease. Further evaluation of this hypothesis requires further experimental data and analysis.