

A Partition Function Algorithm for Interacting Nucleic Acid Strands

Hamidreza Chitsaz^{1,*} Raheleh Salari^{1,*} S. Cenk Sahinalp^{1,†} and Rolf Backofen^{2,†}

¹Lab for Computational Biology, School of Computing Science, Simon Fraser University, Burnaby, BC, Canada

²Bioinformatics, Institute of Computer Science, Albert-Ludwigs-Universität, Freiburg, Germany

ABSTRACT

Recent interests, such as RNA interference and antisense RNA regulation, strongly motivate the problem of predicting whether two nucleic acid strands interact.

Motivation: Regulatory non-coding RNAs (ncRNAs) such as microRNAs play an important role in gene regulation. Studies on both prokaryotic and eukaryotic cells show that such ncRNAs usually bind to their target mRNA to regulate the translation of corresponding genes. The specificity of these interactions depends on the stability of intermolecular and intramolecular base pairing. While methods like deep sequencing allow to discover an ever increasing set of ncRNAs, there are no high-throughput methods available to detect their associated targets. Hence, there is an increasing need for precise computational target prediction. In order to predict base-pairing probability of any two bases in interacting nucleic acids, it is necessary to compute the interaction partition function over the whole ensemble. The partition function is a scalar value from which various thermodynamic quantities can be derived. For example, the equilibrium concentration of each complex nucleic acid species and also the melting temperature of interacting nucleic acids can be calculated based on the partition function of the complex.

Results: We present a model for analyzing the thermodynamics of two interacting nucleic acid strands considering the most general type of interactions studied in the literature. We also present a corresponding dynamic programming algorithm which computes the partition function over (almost) all physically possible joint secondary structures formed by two interacting nucleic acids in $O(n^6)$ time. We verify the predictive power of our algorithm by computing (1) the melting temperature for interacting RNA pairs studied in the literature and (2) the equilibrium concentration for several variants of the OxyS-fhlA complex. In both experiments our algorithm shows high accuracy and outperforms competitors.

Availability: Software and web server is available at

<http://compbio.cs.sfu.ca/taverna/pirna/>

Contact: cenk@cs.sfu.ca and backofen@informatik.uni-freiburg.de

1 INTRODUCTION

Starting with the discovery of microRNAs (miRNAs) and the advent of genome-wide transcriptomics, it has become clear that RNA plays a large variety of important roles in living organisms that extend far beyond being a mere intermediate in protein biosynthesis [32]. Several of these non-coding RNAs (ncRNAs)

regulate gene expression post-transcriptionally through base pairing (and establishing a joint structure) with a target mRNA, as per the eukaryotic miRNAs and small interfering RNAs (siRNAs) [4, 14, 39], antisense RNAs [34, 6] or bacterial small regulatory RNAs (sRNAs) [13]. In addition to such endogenous regulatory ncRNAs, antisense oligonucleotides have been used as exogenous inhibitors of gene expression; antisense technology is now commonly used as a research tool as well as for therapeutic purposes. Furthermore, synthetic nucleic acid systems have been engineered to self assemble into complex structures performing various dynamic mechanical motions [29, 30, 31, 33, 38].

Despite all the above advances, computational methods for predicting ncRNA-target mRNA interactions suffer from a low specificity (see below for an overview); this is possibly due to two technical reasons. First, several of these methods consider restricted versions of the problem (e.g. simplified energy functions or restricted types of interactions) - this is mostly for computational reasons. Second, a quantitative analysis of binding thermodynamics between oligonucleotides and target RNAs is often lacking. To determine the binding effectiveness, an accurate analysis of the thermodynamics of two interacting nucleic acid strands is necessary.

In this paper we aim to compute how likely two RNA or DNA strands are to interact, and to predict base-pairing probability of any two bases - which we then use to quantitatively measure the strength, probability, and stability of the joint structure established by the interacting strands [19]. To correctly calculate those probabilities, it is necessary to compute the partition function over the whole ensemble of possible individual and joint secondary structures. The partition function is a scalar value from which various thermodynamic quantities can be derived [17]. For instance, one can compute the equilibrium concentration of each complex nucleic acid species from their partition functions. Also, the partition function can be used to predict the melting temperature of interacting nucleic acids.

Although algorithms for predicting the most likely (the lowest total free energy) joint structure that can be formed by two interacting RNA strands are available (see, for example, Alkan et al. [1]), previous little work has been done for computing the partition function of interacting RNA strands. It is important to note that designing an algorithm to compute the partition function is more challenging than that to predict the minimum free energy secondary structure: a partition function algorithm should guarantee that every joint structure is considered exactly once.

In this paper we present an $O(n^6)$ time algorithm to compute the partition function over the type of interactions that Alkan et al. [1]

*Joint first author.

†To whom correspondence should be addressed.

considered¹. We extend the standard energy model for a single RNA model to an energy model for the joint secondary structure of interacting strands by considering new types of (joint) structural components. We verify our algorithm (and the associated software we developed) by computing the melting temperature for RNA pairs available [9, 21, 37] and the equilibrium concentration for OxyS-fhlA complexes for wild-type fhlA and 4 other mutants reported in the literature [3]. In both experiments our algorithm shows high accuracy and outperforms existing alternatives.

Related Work

During the last few decades, several computational methods emerged to study the secondary structure thermodynamics of a single nucleic acid strand. Nearest Neighbor Thermodynamic model has become the standard energy model for a nucleic acid secondary structure [20]. The standard energy model is based on the assumption that stacking base pairs and loop entropies contribute additively to the free energy of a nucleic acid secondary structure. More recently, the standard energy model has been extended for pseudoknots [8, 12]. Based on additivity of the energy, efficient dynamic programming algorithms for predicting the minimum free energy secondary structure [24, 36, 40, 28] and computing the partition function of a single strand [22, 12] have been developed.

Some previous attempts to analyze the thermodynamics of multiple interacting strands concatenate input sequences in some order and consider them as a single strand. For example, `pairfold`[2] and `RNAcofold` from Vienna package [5] concatenate the two input sequences into a single strand and predict its minimum free energy structure. Dirks et al. present a method, as a part of `NUPACK`, that concatenates the input sequences in some order, carefully considering symmetry and sequence multiplicities, and computes the partition function for the whole ensemble of complex species[11]. However, concatenating the sequences is not accurate at all as even if pseudoknots are considered, some useful interactions are excluded while many physically impossible interactions are included (for example physically impossible crossing interactions).

Alternatively, several methods avoid internal base-pairing in either strand, and compute the minimum free energy secondary structure for their hybridization under this constraint (`RNAhybrid` [27], `UNAFold` [10, 18], and `RNA duplex` from Vienna package [5]). These approaches naturally work only for simple cases involving typically very short strands.

A third set of methods predict the secondary structure of each individual RNA independently, and predict the (most likely) hybridization between the unpaired regions of the two molecules. More sophisticated alternatives view interaction as a multi step process [23, 35, 7]: 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.

In addition to the above approaches a number of studies aimed to compute the minimum total energy joint structure between two interacting strands under energy models with growing complexity. For instance, Pervouchine devised a dynamic programming

algorithm to maximize the number of base pairs among interacting strands [25]. A followup work by Kato et al. proposed a grammar based approach to RNA-RNA interaction prediction [16]. More generally, Alkan et al. [1] studied the joint secondary structure prediction problem under three different models: 1) base pair counting, 2) stacked pair energy model, and 3) loop energy model. Alkan et al. proved that the general RNA-RNA interaction prediction under all three energy models is an NP-hard problem. Therefore, they suggested some natural constraints on the topology of possible joint secondary structures which are satisfied by all examples of complex RNA-RNA interactions in the literature. The resulting algorithms compute the minimum free energy secondary structure among all possible joint secondary structures that do not contain (internal) pseudoknots, crossing interactions (i.e. external pseudoknots), and *zigzags* (please see section 2.1 for the exact definition).

2 METHODS

2.1 Preliminaries

Throughout this paper, we denote the two nucleic acid strands by **R** and **S**. Strand **R** is indexed from 1 to L_R , and **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. **R** in 5' \rightarrow 3' with **S** in 3' \leftarrow 5' direction. Each nucleotide is paired with at most one nucleotide in the same or the other strand. We refer to the i^{th} nucleotide in **R** and **S** by i_R and i_S respectively. The subsequence from the i^{th} nucleotide to the j^{th} nucleotide in a strand is denoted by $[i, j]$.

An intramolecular base pair between the nucleotides i and j in a strand is called an *arc* and denoted by a bullet $i \bullet j$. An intermolecular base pair between the nucleotides i_R and i_S is called a *bond* and denoted by a circle $i_R \circ i_S$. An arc $i_R \bullet j_R$ covers a bond $l_R \circ k_S$ if $i_R < l_R < j_R$. We call $i_R \bullet j_R$ an *interaction arc* if there is a bond $l_R \circ k_S$ covered by $i_R \bullet j_R$. A *kissing arc* is an interaction arc that directly covers a bond. More precisely, we call $i_R \bullet j_R$ a kissing arc if it covers a bond $l_R \circ k_S$ such that if $i'_R \bullet j'_R$ covers the same bond $l_R \circ k_S$, then $i'_R \leq i_R$ and $j'_R \leq j_R$. A subsequence $[i_R, j_R]$ contains a *direct bond*, $l_R \circ k_S$, if $i_R \leq l_R \leq j_R$ and no arc within $[i_R, j_R]$ covers $l_R \circ k_S$. Assuming $i_R < j_R$, two bonds $i_R \circ i_S$ and $j_R \circ j_S$ are called *crossing bonds* if $i_S < j_S$. An interaction arc $i_R \bullet j_R$ in a strand *subsumes* a subsequence $[i_S, j_S]$ in the other strand if for all bonds $l_R \circ k_S$, if $i_S \leq k_S \leq j_S$ then $i_R < l_R < j_R$. Two interaction arcs are *equivalent* if they subsume one another. Two interaction arcs $i_R \bullet j_R$ and $i_S \bullet j_S$ are part of a *zigzag*, if neither $i_R \bullet j_R$ subsumes $[i_S, j_S]$ nor $i_S \bullet j_S$ subsumes $[i_R, j_R]$.

In this paper, we assume there are no pseudoknots in individual secondary structures of **R** and **S**, and also there are no crossing bonds and zigzags between **R** and **S**.

2.2 Interaction Energy Model

An unspseudoknotted secondary structure s of a single nucleic acid, in the standard energy model [20], is decomposed into loops, and a free energy is associated with every loop in s . The total free energy G_s is the sum of loop free energies. The standard model consists of the following loop types: 1) Hairpin, 2) Interior, which can be stack, bulge, or internal loop, and 3) Multiloop whose energies are denoted by G^{hairpin} , G^{interior} , and G^{multi} respectively.

In an interaction secondary structure of two strands under our assumptions (remember we do not allow pseudoknots, crossing bonds, and zigzags in this work), new kinds of components can appear. We extend the standard energy model by defining those new kinds of interaction components. Similar to the standard case, an interaction secondary structure s can be decomposed into intramolecular loops and the new interaction components such that the total free energy G_s is sum of the free energies of loops and interaction

¹ which is, to our knowledge, the most general type of interactions considered in the literature.

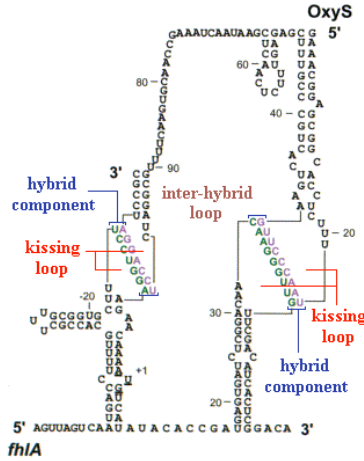


Fig. 1. Interaction components of OxyS-fhlA pair presented in [3].

components. Fig. 1 shows the decomposition of OxyS-fhlA pair secondary structure into the interaction components [3]. These components and their free energy contributions are

- Hybrid: $G_{\{k_R^i \circ k_S^i\}}^{\text{hybrid}}$ is the free energy of a joint secondary structure consisting of a series of bonds, $k_R^i \circ k_S^i, i = 1, \dots, m$, with no intramolecular base pairing or branching. We call such a component *hybrid*. We define the energy associated with a hybrid component by

$$G_{\{k_R^i \circ k_S^i\}}^{\text{hybrid}} = \beta_1 + \sigma \sum_{i=1}^{m-1} G_{k_R^i, k_R^{i+1}, k_S^{i+1}, k_S^i}^{\text{interior}}, \quad (1)$$

in which β_1 is the penalty for the formation of the hybrid, and $\sigma \leq 1$ is the ratio of the free energy of intermolecular to that of intramolecular interior loops (as suggested by [1]). See Fig. 2. Note that with $\beta_1 = 0, \sigma = 1$, G^{hybrid} is identical to the energy proposed by RNAhybrid, first introduced by Rehmsmeier et al. which considers only one hybrid component for mRNA/target duplexes and does not allow any intramolecular structure [27].

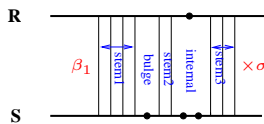


Fig. 2. A hybrid component between the two strands whose free energy is $G^{\text{hybrid}} = \beta_1 + \sigma(G^{\text{stem1}} + G^{\text{bulge}} + G^{\text{stem2}} + G^{\text{internal}} + G^{\text{stem3}})$.

- Kissing: $G_{U^k, B^k}^{\text{kissing}}$ is the energy of an intramolecular loop (hairpin, interior, or multiloop) that makes interaction with the other strand. Such component is called a *kissing loop*. The energy associated with a kissing loop is given by

$$G_{U^k, B^k}^{\text{kissing}} = \beta_2 U^k + \beta_3 B^k, \quad (2)$$

in which B^k is the number of loops and U^k the number of unpaired bases in the kissing loop (Fig. 3). Note in our model we use different β_1 and σ values for a hybrid component covered by a kissing loop.

- Inter-hybrid: $G^{\text{inter-hybrid}}$ is the energy of an intermolecular loop bounded by two bonds belonging to two consecutive hybrid components. Bases in either sequence facing this kind of loop might be the end points of only arcs and not bonds. We call such a

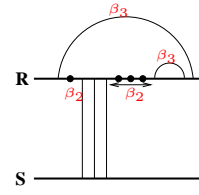


Fig. 3. A kissing loop in R that interacts with the other strand S . In this case, the free energy of the kissing loop is $G^{\text{kissing}} = 4\beta_2 + 2\beta_3$.

component *inter-hybrid loop*. In this work the energy contribution of an inter-hybrid loop is assumed to be zero.

2.3 Interaction Partition Function

The partition function is a weighted sum over the set of all possible secondary structures S

$$Q(T) = \sum_{s \in S} e^{-G_s/RT} \quad (3)$$

where R is the universal gas constant and T is the temperature.

Efficient algorithms for computing the partition function for a single strand have been given. McCaskill gave the first partition function algorithm for a single unpseudoknotted nucleic acid strand [22], and Dirks and Pierce gave a partition function algorithm for a single strand allowing pseudoknots [12]. However, computing the partition function for multiple interacting strands has not been properly addressed. In previous attempts multiple strands are concatenated in some order and partition function for the resulting single strand is computed [10, 5, 11]. That approach is not accurate as even if pseudoknots are considered, some useful interactions are excluded while many physically impossible interactions are included (for example physically impossible crossing interactions). On the other hand, considering all possible secondary structures makes the problem NP-hard [1]. Therefore, we only consider those secondary structures that do not contain pseudoknots, crossing bonds, and zigzags.

Interaction Partition Function (IPF) Problem

Given a pair of nucleic acid strands R and S , and a temperature T , compute the partition function, $Q^I(T)$, over S^I the set of all possible single or duplex secondary structures that do not contain pseudoknots, crossing bonds, and zigzags.

Input: nucleic acid strands R and S .

Output:

$$Q^I(T) = \sum_{s \in S^I} e^{-G_s/RT}.$$

We give a recursive algorithm, called Partition function for InteRacting Nucleic Acids (piRNA), for the IPF problem. In all of our recursions, the considered cases are disjoint. This fact shows that every possible secondary structure is reached by exactly one trajectory in the recursion process. Our algorithm guarantees to consider all possible secondary structures exactly once. Since our algorithm covers so many cases, we do not include all the details here. A comprehensive description of our algorithm is available in our supplementary materials.

We present our algorithm using recursion diagrams [12, 28]. Our algorithm computes two types of recursive quantities: 1) the partition function of a subsequence $[i, j]$ in one strand, and 2) the joint partition function of subsequences $[i_R, j_R]$ and $[i_S, j_S]$. A *region* is the domain over which a partition function is computed. *Terminal bases* are the boundaries of a region. For the first type, region is $[i, j]$ with i and j terminal bases. For the second type, region is $[i_R, j_R] \times [i_S, j_S]$ with i_R, j_R, i_S , and j_S terminal bases. The *length pair* of region $[i_R, j_R] \times [i_S, j_S]$ is $(l_R =$

$j_R - i_R + 1, l_S = j_S - i_S + 1$). Our algorithm starts with $(l_R = 1, l_S = 1)$ and considers all length pairs incrementally up to $(l_R = L_R, l_S = L_S)$. For a fixed length pair (l_R, l_S) , recursive quantities for all the regions $[i_R, i_R + l_R - 1] \times [i_S, i_S + l_S - 1]$ are computed.

For computing the partition function of a subsequence in one strand we use McCaskill's algorithm [22]. McCaskill's algorithm is shown in Fig. 4, in which $Q_{i,j}$ is the partition function for the subsequence $[i, j]$. Throughout this paper, a horizontal line indicates the phosphate backbone, a solid curved line indicates an arc, and a dashed curved line encloses a region and denotes its two terminal bases which may be paired or unpaired. Letter(s) within a region specify a recursive quantity. White regions are recursed over and blue regions indicate those portions of the secondary structure that are fixed at the current recursion level and contribute their energy to the partition function as defined by the energy model. Green and red regions have the same recursion cases as the corresponding white regions, except that for the green regions multiloop energy and for red regions kissing loop energy is applied, i.e. the corresponding penalties for each unpaired base and base pair should be applied.

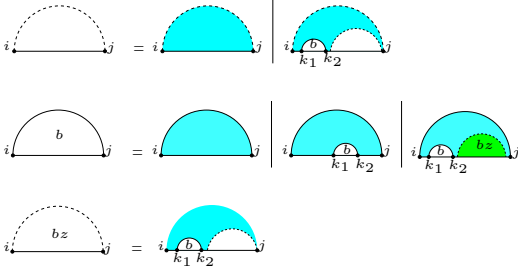


Fig. 4. McCaskill's algorithm: recursion for $Q_{i,j}$, the partition function for the subsequence $[i, j]$. Above, $Q_{i,j}^b$ is the partition function for the subsequence $[i, j]$ assuming i and j are base paired, and $Q_{i,j}^{bz}$ is the partition function for the subsequence $[i, j]$ assuming there is at least one arc in the region.

In Fig. 4, the first case of $Q_{i,j}$ corresponds to an empty structure (that constitutes no base pairs) whose free energy is assumed to be zero, thus its contribution to the partition function is $e^{-G_{i,j}^{empty}/RT} = 1$. In the other case, there exists at least one arc and the leftmost one is $k_1 \bullet k_2$. It contributes $Q_{k_1, k_2}^b Q_{k_2+1, j}$ to the partition function, therefore,

$$Q_{i,j} = 1 + \sum_{i \leq k_1 < k_2 \leq j} Q_{k_1, k_2}^b Q_{k_2+1, j}. \quad (4)$$

The second line shows the cases of $Q_{i,j}^b$ which is the partition function for the subsequence $[i, j]$ assuming i and j are base paired. The arc $i \bullet j$ can close different substructures: hairpin, interior, or multiloop. The energy contribution of each substructure is calculated based on the standard thermodynamics energy model. The third line shows cases of $Q_{i,j}^{bz}$ which is the partition function for the subsequence $[i, j]$ assuming the region constitutes at least one arc. A region tagged by bz and colored by green is contained in a multiloop and the penalty of multiloop should be applied to it. Explicit equations for $Q_{i,j}^b$ and $Q_{i,j}^{bz}$ are given in the supplementary material.

In the following, we present all cases of Q_{i_R, j_R, i_S, j_S}^I which is the interaction partition function for the region $[i_R, j_R] \times [i_S, j_S]$. A solid vertical line indicates a bond, a dashed vertical line denotes two terminal bases of a region which may be base paired or unpaired, and a dotted vertical line denotes two terminal bases of a region which are assumed to be unpaired. For the interaction partition functions, grey regions indicate a reference to the partition functions for the single sequences. Fig. 5 shows the cases of Q^I : 1) there is no bond between the two subsequences, 2) the

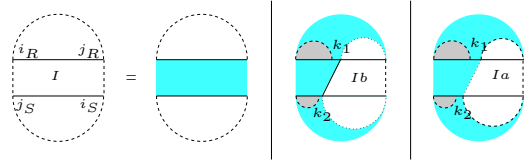


Fig. 5. Cases of the interaction partition function Q_{i_R, j_R, i_S, j_S}^I . The first case constitutes no bonds. In the second case, the leftmost bond is a direct bond on both subsequences. In the third case, the leftmost bond is covered by an interaction arc in at least one subsequence.

leftmost bond is a direct bond in both subsequences, and 3) the leftmost bond is covered by an arc in at least one subsequence. Therefore,

$$Q_{i_R, j_R, i_S, j_S}^I = Q_{i_R, j_R} Q_{i_S, j_S} + \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} Q_{i_R, k_1-1} Q_{k_2+1, j_S} Q_{k_1, j_R, i_S, k_2}^{Ib} + \sum_{\substack{i_R \leq k_1 < j_R \\ i_S < k_2 \leq j_S}} Q_{i_R, k_1-1} Q_{k_2+1, j_S} Q_{k_1, j_R, i_S, k_2}^{Ia}, \quad (5)$$

in which $Q_{k_1, j_R, i_S, k_2}^{Ib}$ is the interaction partition function for the region $[k_1, j_R] \times [i_S, k_2]$ assuming $k_1 \bullet k_2$ is a bond, and $Q_{k_1, j_R, i_S, k_2}^{Ia}$ is the interaction partition function for the region $[k_1, j_R] \times [i_S, k_2]$ assuming that the leftmost bond in the region is covered by an arc in at least one subsequence. Figs. 6, 8 show the recursion for Q^{Ib} and Q^{Ia} where b stands for bond and a stands for arc.

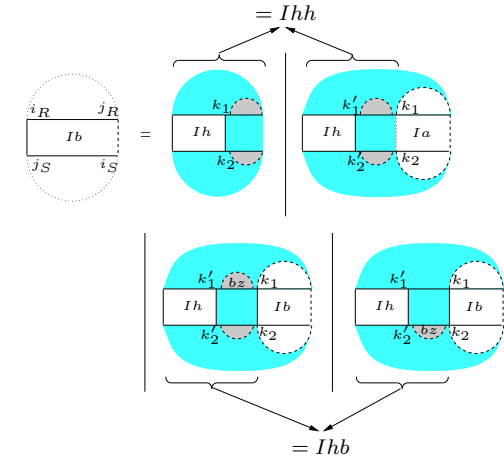


Fig. 6. Recursion for $Q_{i_R, j_R, i_S, j_S}^{Ib}$ assuming $i_R \circ j_S$ is a bond. We show a version of the recursion that contains two split points in each sequence for simplicity reasons. However, this would increase the complexity and can easily be resolved by introducing two additional matrices Q^{Ihh} and Q^{Ihb} for the region $[i_R, k_1] \times [k_2, j_S]$ as indicated by the arrows (see supplementary materials for a full definition).

Fig. 6 shows the recursion for $Q_{i_R, j_R, i_S, j_S}^{Ib}$, the interaction partition function for the region $[i_R, j_R] \times [i_S, j_S]$ assuming $i_R \circ j_S$ is a bond. Since we have a β_1 penalty for each hybrid component, the recursion for Q^{Ib} has to determine whether the region contains one or several hybrid components. In all cases, Q^{Ih} contains the full hybrid component containing the bond $i_R \circ j_S$ (see Fig. 7 for Q^{Ih} recursion). The first possibility reflects the case where we have only one hybrid component. In the other cases, we

have always at least two hybrid components. The subsequent intermolecular bond starts a new hybrid component iff 1) it is not direct in at least one subsequence, i.e. it is covered by an arc in the associated regions (case 2 of the Q^{Ib} recursion), or 2) there is at least one arc between the two successive intermolecular bonds (case 3 and 4 of the Q^{Ib} recursion). Using the additional matrices Q^{Ihh} and Q^{Ihb} , we get

$$Q_{i_R, j_R, i_S, j_S}^{Ib} = Q_{i_R, j_R, i_S, j_S}^{Ihh} + \sum_{\substack{i_R < k_1 < j_R \\ i_S < k_2 < j_S}} Q_{i_R, k_1, k_2, j_S}^{Ihb} Q_{k_1, j_R, i_S, k_2}^{Ib} + \sum_{\substack{i_R < k_1 < j_R \\ i_S < k_2 < j_S}} Q_{i_R, k_1, k_2, j_S}^{Ihh} Q_{k_1, j_R, i_S, k_2}^{Ia}. \quad (6)$$

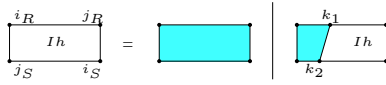


Fig. 7. Cases of $Q_{i_R, j_R, i_S, j_S}^{Ih}$ the interaction partition function for a single hybrid component.

Fig. 7 shows the cases of Q^{Ih} : 1) there is no bond other than $i_R \circ j_S$ and $i_S \circ j_R$ in the region, and 2) there exist more bonds between $i_R \circ j_S$ and $i_S \circ j_R$, the leftmost of which is $k_1 \circ k_2$.

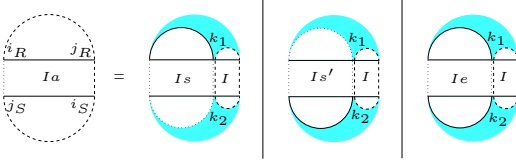


Fig. 8. Cases of $Q_{i_R, j_R, i_S, j_S}^{Ia}$ for which we assume at least one of i_R and j_S is the end point of an interaction arc.

Fig. 8 shows the cases of $Q_{i_R, j_R, i_S, j_S}^{Ia}$ for which at least one of i_R and j_S is the end point of an interaction arc: 1) $i_R \bullet k_1$ subsumes $[k_2, j_S]$ and k_2 is not base paired with j_S , 2) $k_2 \bullet j_S$ subsumes $[i_R, k_1]$ and i_R is not base paired with k_1 , and 3) $i_R \bullet k_1$ and $k_2 \bullet j_S$ are equivalent. If just one of i_R and j_S is the end point of an interaction arc while the other one is the end point of a bond, then the interaction arc subsumes the other subsequence. If both i_R and j_S are end points of interaction arcs, then one of the arcs subsumes the other one or they are equivalent. Therefore,

$$Q_{i_R, j_R, i_S, j_S}^{Ia} = \sum_{\substack{i_R < k_1 \leq j_R \\ i_S \leq k_2 \leq j_S}} Q_{i_R, k_1, k_2, j_S}^{Is} Q_{k_1+1, j_R, i_S, k_2-1}^I + \sum_{\substack{i_R \leq k_1 \leq j_R \\ i_S < k_2 \leq j_S}} Q_{i_R, k_1, k_2, j_S}^{Is'} Q_{k_1+1, j_R, i_S, k_2-1}^I + \sum_{\substack{i_R < k_1 \leq j_R \\ i_S < k_2 \leq j_S}} Q_{i_R, k_1, k_2, j_S}^{Ie} Q_{k_1+1, j_R, i_S, k_2-1}^I, \quad (7)$$

in which $Q_{i_R, k_1, k_2, j_S}^{Is}$ is the interaction partition function of $[i_R, k_1] \times [k_2, j_S]$ assuming $i_R \bullet k_1$ is an interaction arc that subsumes $[k_2, j_S]$, $Q_{i_R, k_1, k_2, j_S}^{Is'}$ is the symmetric counterpart of $Q_{i_R, k_1, k_2, j_S}^{Is}$, and $Q_{i_R, k_1, k_2, j_S}^{Ie}$ is the interaction partition function of $[i_R, k_1] \times [k_2, j_S]$ assuming $i_R \bullet k_1$ and $k_2 \bullet j_S$ are equivalent interaction arcs.

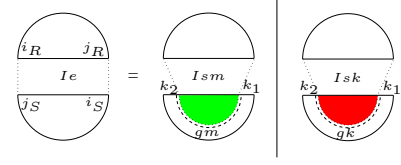


Fig. 9. Cases of $Q_{i_R, j_R, i_S, j_S}^{Ie}$, for which $i_R \bullet j_R$ and $i_S \bullet j_S$ are equivalent interaction arcs.

For Q^{Ie} , it does not make any difference which one of the covering arcs $i_R \bullet j_R$ and $i_S \bullet j_S$ is extracted first. We first extract the covering arc from **S** (see Fig. 9). Extracting the covering arc, the remaining subsequence of **S** contains either at least one direct bond, in which case kissing loop penalty should be applied, or multiple interaction arcs, in which case multiloop penalty should be applied. Hence, Fig. 9 is appropriately colored by green and red to remind the type of penalty.

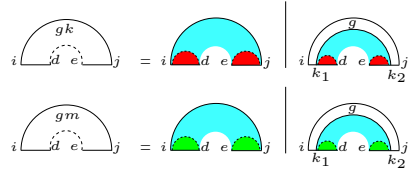


Fig. 10. Recursion for $Q_{i, d, e, j}^{gk}$ and $Q_{i, d, e, j}^{gm}$ the partition functions for $[i, j]$ excluding the gap $[d, e]$, assuming i and j are base paired. The gap in Q^{gk} is assumed to contain a direct bond hence the red color, and the gap in Q^{gm} is assumed to contain multiple interaction arcs hence the green color. In both cases, $i < k_1 \leq d$ and $e \leq k_2 < j$.

Note that $Q_{i, d, e, j}^{gk}$ and $Q_{i, d, e, j}^{gm}$ are the partition functions for $[i, j]$ excluding the gap $[d, e]$ assuming i and j are base paired. For Q^{gk} , the gap is assumed to contain a direct bond, and for Q^{gm} the gap is assumed to contain multiple interaction arcs (see Fig. 10). The only difference between Q^{gk} and Q^{gm} is in the penalty type. For both Q^{gk} and Q^{gm} , there are two cases: 1) there is no more spanning interaction arc in the region, and 2) there is at least another innermost spanning interaction arc $k_1 \bullet k_2$. In both groups, there could be some additional intramolecular structure in the region. The quantity $Q_{i, k_1, k_2, j}^g$ is the partition function for the subsequence $[i, j]$ excluding the gap $[k_1, k_2]$ assuming $i \bullet j$ and $k_1 \bullet k_2$. The gap partition function Q^g is similar to Q^g in Dirks-Pierce's algorithm [12]. Please see our supplementary materials for the details of Q^{gk} , Q^{gm} , and Q^g .

The union of the cases of Q^{Isk} and Q^{Ism} comprises the cases of Q^{Is} . Similar to the cases of Q^{Ie} , we extract the covering arc from Q^{Isk} and Q^{Ism} to obtain Q^{Imm} , Q^{Imk} , Q^{Ikm} , and Q^{Ikk} , where k stands for kissing (or equivalently containing a direct bond) and m for multiple interaction arcs. Note that all four terminal bases of the region of these four quantities are paired, i.e. each terminal base is either the end point of a bond or of an interaction arc. These four quantities have complicated cases. Due to lack of space, we explain them in our supplementary materials.

3 EXPERIMENTS

Here, we report our implementation of the algorithm and two types of experiments we performed to test the predictive power of our algorithm:

(1) Predicting the melting temperature of RNA duplexes is an important application of the partition function for interacting nucleic

acid pairs [10]; our first experiment tests how accurately our algorithm predicts the melting temperature of RNA pairs collected from several sources in the literature with respect to the accuracy of available alternatives, `RNAcofold` from Vienna package v1.7.2 [5] and `UNAFold v3.6` which is a new version of former `mfold` [18]. We remind the reader that `RNAcofold` concatenates the two RNA strands and computes the partition function for the resulting single strand. Therefore, it does not consider many cases that our algorithm considers. `UNAFold v3.6`, on the other hand, simplifies the problem by forbidding intramolecular base pairing. It computes the partition function of the two strands over just hybridization structures. As can be expected, our algorithm consistently outperforms the alternatives in all three data sets.

(2) A novel experiment (which, to our knowledge has not been performed successfully by any other program to date) uses our algorithm to predict the equilibrium concentration of an RNA-RNA complex, in particular the OxyS-fhlA interaction [3].² We successfully predicted the equilibrium concentrations for OxyS with wild-type fhlA and 4 other fhlA mutants.

Note that the parameters used by our program in the above experiments have been manually optimized as computational learning methods for fine tuning the parameters require prohibitive computational resources. It may be possible to improve the accuracy of our program through a better selection of parameters.

3.1 Implementation

We remind the reader that the time and space complexity of our algorithm are $O(n^6)$ and $O(n^4)$ respectively; here $n = \max(L_R, L_S)$ is the maximum length of the two input strands. We implemented our algorithm in C++, and used the energy functions and energy parameters of `UNAFold v3.6` for a single strand [18]. The parameters used by our program for our own interaction energy model are given in the next section. As per Section 2.2 we use a different β_1 penalty and σ for a hybrid component that is covered by a kissing loop. The parameters for a hybrid component that is not covered by a kissing loop is denoted by β'_1 and σ' . We add an AU penalty to the energy of a hybrid component per each terminal AU base pair; this penalty is motivated by [37]. Similar to `RNAhybrid`, the interior loops in a hybrid component are restricted to a constant maximum length, in either sequence, which is set to 15 in this work.

Since our algorithm considers many more possible secondary structures in comparison to alternative methods, our program has a higher running time. Fortunately, our algorithm can be easily parallelized as the dynamic programming tables computed by our program on subsequence pairs depend only on their (proper) subregions. We parallelized our program using OpenMP 3.0. Our experiments were performed on a large scale shared memory parallel platform with 64 PPC 1.9 GHz processors with 256 GB RAM. We ran our program for strands of length between 5 nt to 120 nt. The running time of our program for short strands (~ 20 nt) was

less than 1 minute - for longer strands (~ 120 nt) it was about 10 hours.

3.2 Data Sets

The first data set that we used for predicting melting temperature contains all 9 different RNA pairs reported in Table 3 of [37]. It contains almost complementary 5-7nt RNA pairs that were designed to optimize the thermodynamic parameters for terminal base pairs. Their melting temperatures vary from 29.8°C to 53.7°C .

The second data set that we used for computing melting temperature contains all 12 different RNA pairs reported in Table 1 of [9]. These RNA pairs are designed to optimize the thermodynamic parameters for three-way multi loops. In each pair of this data set, the first RNA has ~ 20 nt and the second one has ~ 10 nt. The experimental melting temperatures were determined from heat absorbance measurements by two different methods which are explained as "Method 3" and "Method 4" in [26]. Although these pairs are very similar, the average difference of the two methods for this data set is 2.49°C . This suggests that there may exist RNA pairs with exceptional features in this set.

The third data set that we used for computing melting temperature contains all 62 different RNA pairs reported in Tables 3 and 4 of [21]. These pairs are designed to optimize the thermodynamic parameters for three- and four-way multi loops. In each pair of this data set, the first RNA has 22-40nt and the second one has 10-14nt. Again, the experimental melting temperatures were determined by two different methods. This data set is large enough with longer sequences, and the average difference of the two methods for this data set is 0.7°C , smaller than that for the second data set. Moreover, the variance and maximum of the difference is smaller than those of the second data set. Overall, this data set is more reliable than the previous one. These three data sets are all we were able to collect from the literature.

3.3 Melting Temperature

As mentioned before, predicting the melting temperature of RNA duplexes is one of the most important applications of the partition function for interacting nucleic acid pairs [10]. Table 1 shows the melting temperatures computed by our program, `RNAcofold`, and `UNAFold v3.6` for the first data set. In this set, the strands are short, and as we expected, our algorithm is highly accurate with only 1.48°C absolute difference from experimental values on average. It can be seen that `RNAcofold` and `UNAFold` perform relatively poorly, and their predicted melting temperatures differ from the experimental values by about 9°C on average.

Table 2 shows the melting temperatures predicted by the three programs for the second data set. Each pair is referred to by an identifier (A, B, \dots, L) . Please refer to our supplementary material or [9] to see the exact sequences of each pair. As mentioned before, the experimental melting temperatures were determined from heat absorbance measurements by two different methods which are explained as "Method 3" and "Method 4" in [26]. We refer to the melting temperature values computed by "Method 3" and "Method 4", by T_c and T_l respectively. `RNAcofold` accuracy obviously dropped in this case, whereas `UNAFold` accuracy did not change much in comparison to the results for the first data set. The accuracy of our method has also dropped a bit, which may be because of some RNA pairs with exceptional features.

² Equilibrium concentrations of another complex formed by CopA/I-CopT is also available in the literature [15], however the interaction has tertiary structural components, i.e. a very long pair of kissing hairpins forming a helix, anti-helix pair with a long gap in between. Alkan et al. [1] were able to establish the most likely joint structure between this RNA pair only through postprocessing. This complex requires some additional constraints on the lengths of interacting loops which are not incorporated into our model due to additional computational complexity they would impose.

Table 1. Experimental and predicted melting temperatures for the first data set (see Section 3.2 and [37]).

Pairs	Experiment	piRNA	RNAfold	UNAFold
ACGCA/UGCGU	29.8	29.41	42.64	46.14
GCACG/CGUGC	37.5	36.07	46.61	43.91
AGCGA/UCGCU	30.2	30.38	42.68	45.15
GCUCG/CGAGC	37.2	36.88	47.75	44.71
ACUGUCA/UGACAGU	48.2	44.91	56.8	57.59
GUCACUG/CAGUGAC	51.1	49.4	58.44	55.91
AGUCUGA/UCAGACU	45.7	45.47	56.4	56.68
GACUCAG/CUGAGUC	52	49.96	59.11	56.25
GAGUGAG/CUCACUC	53.7	49.97	59.07	56.00
Avg. error		1.48	9.35	8.55
Spearman rank corr.		0.97	0.97	0.57

All values, except Spearman's rank correlation, are in °C.

Table 2. Experimental and predicted melting temperatures for the set of RNA pairs reported in [9].

Pairs	Experiment		piRNA	RNAfold	UNAFold
	T_l	T_c			
A	28.7	30.3	32.44	50.99	21.52
B	19	20.5	31.55	52.55	33.22
C	33.6	33.6	32.94	53.11	39.77
D	33.9	36	32.43	51.02	26.85
E	23	24.4	31.66	52.48	32.22
F	34.9	36.9	33.28	54.7	39.91
G	32.4	33.6	32.76	49.76	64.27
H	16.1	18.9	36.41	57.92	29.76
I	29	32.3	32.32	50.99	29.18
J	32.3	37.1	37.01	56.92	28.8
K	23.4	30.7	31.45	49.36	26.18
L	33.5	35.4	32.61	50.51	28.01
Avg. difference	T_l	T_l T_c	T_l T_c	T_l T_c	T_l T_c
	2.49	5.53 4.19	24.21 21.72	8.86 9.38	
Spearman rank corr.	0.87	0.36 0.45	-0.05 0	0.16 0.03	

Each pair is referred to by an identifier (A, B, ..., L). Please refer to our supplementary material or [9] to see the exact sequences of each pair. All values, except Spearman's rank correlation, are in °C.

Table 3 presents the melting temperatures predicted by the three programs for the third data set. As you can see, our program has high accuracy and performs significantly better than RNAfold and UNAFold for this data set. As we argued before, the third data set is the largest and the most reliable of the three data sets. It is important to note that RNAfold and UNAFold both perform poorly either in this case or the two previous cases. Therefore, neither RNAfold nor UNAFold are as reliable as our program for melting temperature prediction.

The running time of our program for the first data set was about a few seconds, for the second data set about 10 minutes, and for the third data set about 72 hours on a Linux PC with Pentium-D 3.6 GHz CPU and 4GB of RAM. Note that we did not use any learning methods for tuning our 6 interaction energy parameters because of the running time of our program. Our interaction energy parameters in melting temperature experiments are $\beta_1 = 5.1$, $\beta_2 = \beta_2 = 0.1$, $\sigma = 0.92$, $\beta'_1 = 4.1$, $\sigma' = 0.95$, which were manually optimized using only the first data set. The second and the third data sets were used as test sets.

3.4 Equilibrium Concentration

Our second set of experiments, to the best of our knowledge, have not been successfully performed by the use of any available program

Table 3. Experimental and predicted melting temperatures for the set of RNA pairs reported in [21].

Pairs	Experiment		piRNA	RNAfold	UNAFold
	T_l	T_c			
G-GC-G/C-C	45.4	46	56.81	37	21.4
G-GC-G/CaC	51.8	52.2	56.84	37	27.15
G-GC-G/Ca ₂ C	55.9	56	56.86	37	27.12
G-GC-G/Ca ₃ C	58.4	57.3	56.85	37	25.73
G-GC-G/CauaC	57.3	56.9	56.84	37	24.35
G-GC-G/Ca ₄ C	56.7	57.1	56.85	37	25.06
GaGC-G/C-C	51.2	49.3	56.94	37.1	21.25
GaGC-G/CaC	55.2	54.7	56.96	41	21.44
GaGC-G/Ca ₂ C	56.1	55.6	56.98	41	21.46
GaGC-G/Ca ₃ C	56.1	54.9	56.97	41	21.44
GaGC-G/CauaC	54.8	54.3	56.98	37.06	21.47
GaGC-G/Ca ₄ C	55.3	54.8	56.98	37	21.39
Ga ₂ GC-G/C-C	55.1	52.9	57	50.17	36.04
Ga ₂ GC-G/CaC	57	56.4	57.03	48.01	36.8
Ga ₂ GC-G/Ca ₂ C	55.6	55.4	57.05	41.84	36.91
Ga ₂ GC-G/Ca ₃ C	55	54.7	57.03	41.84	36.19
Ga ₂ GC-G/CauaC	55.3	54.5	57.3	52.17	36.74
Ga ₂ GC-G/Ca ₄ C	54.1	53.9	57.05	48.01	34.22
Ga ₂ GCaG/C-C	56.6	56.6	57.18	47.01	36.01
Ga ₂ GCaG/CaC	58.7	58.9	57.18	44.1	36.81
Ga ₂ GCaG/Ca ₂ C	58	58.8	57.2	44.1	36.13
Ga ₂ GCaG/Ca ₃ C	56.5	57.5	57.15	44.1	36.96
Ga ₂ GCaG/CauaC	57.2	56.9	57.48	43	35.92
Ga ₂ GCaG/Ca ₄ C	57.9	57.9	57.17	44.1	34.66
Ga ₂ GCa ₂ G/C-C	56	56.9	57.19	37.17	36.14
Ga ₂ GCa ₂ G/CaC	58.7	59.1	57.2	44.1	36.94
Ga ₂ GCa ₂ G/Ca ₂ C	59.7	59.6	57.19	44.1	36.22
Ga ₂ GCa ₂ G/Ca ₃ C	58.6	58.7	57.16	44.1	35.89
Ga ₂ GCa ₂ G/CauaC	57	57.3	57.74	37	35.03
Ga ₂ GCa ₂ G/Ca ₄ C	57.5	58.1	57.18	44.1	34.93
G-UA-G/C-C	50.4	50.8	56.82	46.26	21.53
G-UA-G/CaC	54.3	55.8	57.88	61.42	34.47
G-UA-G/Ca ₂ C	56.6	57.8	57.89	61.42	41.68
G-UA-G/Ca ₃ C	57.6	58.5	57.88	61.42	40.84
G-UA-G/CauaC	57.9	58.7	57.87	61.41	40.96
G-UA-G/Ca ₄ C	58.6	58.5	57.88	61.43	40.64
GaUA-G/C-C	51.6	51.8	56.96	49.18	21.42
GaUA-G/CaC	55.6	55.7	57.01	37.07	30.98
GaUA-G/Ca ₂ C	56.7	57.4	57.04	50.31	31.46
GaUA-G/Ca ₃ C	56.8	56.9	57	44.17	29.91
GaUA-G/CauaC	57	57.1	56.99	37.07	29.98
GaUA-G/Ca ₄ C	56.8	56.8	57.01	50.31	29.29
G-CG-GC-G/C-C	64.8	65.2	57.24	37	21.38
G-CG-GC-G/CaC	58.8	60.4	57.22	37	21.44
G-CG-GC-G/Ca ₂ C	55.6	56.4	57.35	37	21.38
G-CG-GC-G/Ca ₃ C	55.4	55.3	57.32	37	21.56
G-CG-GC-G/Ca ₄ C	53.9	53	57.19	37	21.38
GaCG-GC-G/C-C	57.3	58.7	57.2	37	21.71
GaCG-GC-G/CaC	59.7	61.2	57.21	37	21.76
GaCG-GC-G/Ca ₂ C	55.4	57.2	57.19	37	21.45
GaCG-GC-G/Ca ₃ C	55.2	56.5	57.11	37	21.42
GaCG-GC-G/CauaC	55.2	55.8	57.09	37	21.38
GaCG-GC-G/Ca ₄ C	55	55.3	57.14	37	21.47
GaCG-GCaG/C-C	58.1	58.8	56.9	37	21.54
GaCG-GCaG/CaC	59.3	59.7	56.99	37	21.76
GaCG-GCaG/Ca ₂ C	57.5	59.4	56.89	37	63.08
GaCG-GCaG/Ca ₃ C	57.9	58.2	56.95	37	21.44
GaCG-GCaG/CauaC	58.9	58.3	56.93	37	21.53
GaCG-GCaG/Ca ₄ C	57.3	58.1	56.84	37	21.46
Ga ₂ CGa ₂ GCa ₂ G/C-C	54.4	55.5	57.12	47.17	67.28
Ga ₂ CGa ₂ GCa ₂ G/CaC	55	56.6	57.04	44.01	67.23
Ga ₂ CGa ₂ GCa ₂ G/Ca ₂ C	55.3	57.2	57.12	51.31	66.09
Avg. difference	T_l	T_l T_c	T_l T_c	T_l T_c	T_l T_c
	0.7	1.87 1.95	14.27 14.41	26.5 26.56	
Spearman rank corr.	0.92	0.25 0.35	-0.04 -0.04	0.2 0.28	

Each pair is referred to by an identifier. Please refer to our supplementary material or [21] to see the exact sequences of each pair. All values, except Spearman's rank correlation, are in °C.

to date. Here we predict the equilibrium concentrations for OxyS with wild-type fhlA and 4 other fhlA mutants. OxyS is a small untranslated RNA (109 nt) that is induced in response to oxidative stress in *E. coli*. It acts as a regulator affecting the expression of multiple genes. In particular, OxyS represses the translation of fhlA, a transcriptional activator for formate metabolism, by binding to it. Argaman and Altuvia carried out a series of experiments to measure equilibrium dissociation constants for OxyS with wild-type fhlA and its mutants [3]. To measure the equilibrium dissociation constants, they measured the concentration of OxyS-fhlA complex for a fixed initial OxyS concentration (2nM) and various initial concentrations of fhlA. Their plots are reported in Figure 8 and Table 2 of [3]. Those plots can be predicted from the partition functions for OxyS, fhlA, OxyS-OxyS, fhlA-fhlA, and OxyS-fhlA. To validate our algorithm, we computed these partition functions using our program, and predicted the equilibrium concentrations of OxyS-fhlA complex. Our results are compatible with experimental measurements, as we had expected.

Fig. 11 shows the experimental measurements and our results. Interestingly, our algorithm predicted the equilibrium concentration of OxyS-fhlA complex quite accurately for the wild-type fhlA and all of its mutants. We also experimented with `RNAcofold` and `UNAFold` in this case. Both `RNAcofold` and `UNAFold` predict that the percentage of OxyS in complex is approximately 0 in all five cases for the considered fhlA concentrations. This is probably not very surprising as correctly predicting the equilibrium concentrations is a very difficult task and is highly sensitive to the accuracy of the partition functions. We describe below how to compute the concentrations from partition functions, and why it is difficult to correctly predict those equilibrium concentrations.

Given two nucleic acid strands **R** and **S**, we can compute the equilibrium concentrations of **R**, **S**, **RR**, **SS**, **RS** species, denoted by N_R , N_S , N_{RR} , N_{SS} , N_{RS} respectively, from their partition functions [10]. In the equilibrium, the free energy of a closed system at constant temperature, volume, and pressure tends toward a minimum [17]. Equilibrium concentrations are computed from the chemical equilibrium constants

$$\begin{aligned} K_R &= \frac{Q_{RR}^I}{Q_R^2} = \frac{N_{RR}}{N_R^2}, \\ K_S &= \frac{Q_{SS}^I}{Q_S^2} = \frac{N_{SS}}{N_S^2}, \\ K_{RS} &= \frac{Q_{RS}^I}{Q_R Q_S} = \frac{N_{RS}}{N_R N_S}, \end{aligned} \quad (8)$$

under the constraint $N_{RS} = N_R^0 - 2N_{RR} - N_R = N_S^0 - 2N_{SS} - N_S$, in which N^0 are the initial concentrations of single strands. We noticed that Q_R and Q_S computed by the three programs are very close because they use the same algorithm for a single strand (i.e. McCaskill's). Therefore based on (8), a method can compute equilibrium concentrations correctly only if it computes each individual Q^I accurately. As one can observe in Fig. 11, our program has been able to predict OxyS-fhlA complex concentrations accurately, thus we can conclude that our program computes all Q^I accurately.

As mentioned above, the parameters used by our program on this data set have been manually optimized. Our energy parameters in

this experiment are $\beta_1 = 6.6, \beta_2 = \beta_3 = 0.1, \sigma = 0.9, \beta'_1 = 4.5, \sigma' = 0.9$.

4 CONCLUSION AND FUTURE WORK

In this paper we present `piRNA`, an efficient algorithm to compute a partition function of two interacting nucleic acid strands. Our algorithm considers almost all physically possible secondary structures that do not contain pseudoknots, crossing interactions, and "zigzag"s. In order to specify the free energy of a joint structure established by interacting strands, we extend the standard nearest neighbor single-strand thermodynamic energy model to an energy model for two interacting strands by introducing three new components: (1) hybrid component, (2) kissing loop, and (3) inter-hybrid loop which are modified versions of hybridization, multi loop, and pseudoknot energy models. We verified our algorithm by computing the melting temperature for RNA pairs available in the literature and the equilibrium concentration for OxyS-fhlA complex. In both experiments our algorithm provides high accuracy and outperforms available alternatives.

We computed the melting temperature for RNA pairs in [37, 9, 21] (Tables 1, 2, and 3). On average, the predicted melting temperature by our program is about 2°C different from experimental values. Our program is more than 10°C more accurate than the alternatives, `RNAcofold` and `UNAFold`, on average. It is important to note that `RNAcofold` and `UNAFold` both perform poorly in at least one of the three data sets, while our program is consistently accurate across all three data sets. Therefore, neither `RNAcofold` nor `UNAFold` are as reliable as our program for melting temperature prediction. In addition, our algorithm is able to compute the OxyS-fhlA complex equilibrium concentrations for wild-type and mutated fhlA accurately. Both `RNAcofold` and `UNAFold` predict those equilibrium concentrations to be approximately 0, which does not even roughly follow the experimental measurements.

Although our algorithm is fairly efficient, improving the generality and complexity of our algorithm will be one of our priorities in the near future. In particular we aim to explore whether it is possible to cover more general interactions without increasing the computational complexity of the algorithm.

ACKNOWLEDGEMENT

H. Chitsaz received funding from Combating Infectious Diseases (BCID) initiative. R. Salari was supported by MITACS Research Grant. S.C. Sahinalp was supported by Michael Smith Foundation for Health Research Career Award. R. Backofen received funding from the German Research Foundation (DFG grant BA 2168/2-1 SPP 1258), and from the German Federal Ministry of Education and Research (BMBF grant 0313921 FRISYS).

REFERENCES

- [1]C. Alkan, E. Karakoc, J. H. Nadeau, S. C. Sahinalp, and K. Zhang. RNA-RNA interaction prediction and antisense RNA target search. *Journal of Computational Biology*, 13(2):267–282, 2006.
- [2]M. Andronescu, Z. Zhang, and A. Condon. Secondary structure prediction of interacting RNA molecules. *J. Mol. Biol.*, 345:987–1001, Feb 2005.
- [3]L. Argaman and S. Altuvia. fhlA repression by OxyS RNA: kissing complex formation at two sites results in a stable antisense-target RNA complex. *J. Mol. Biol.*, 300:1101–1112, Jul 2000.

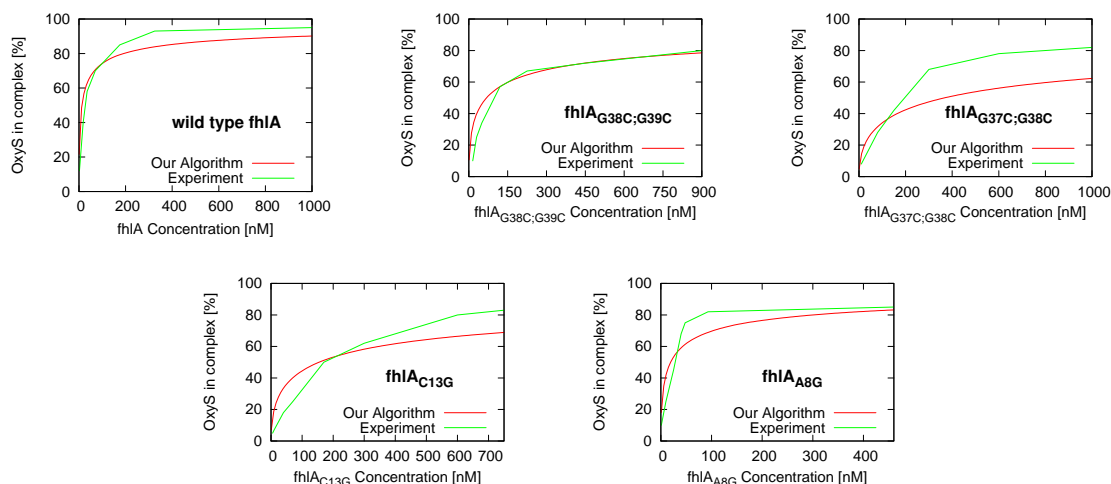


Fig. 11. Experimental and computational determination of equilibrium constants for pairs of OxyS with wild-type and mutated fhlA. Horizontal axis denotes the initial concentration of fhlA, and the vertical axis denotes the percentage of OxyS in OxyS-fhlA complex. Initial concentration of OxyS was $2 \times 10^{-9} M$ [3]. Both RNAfold and UNAFold predict that the percentage of OxyS in complex is approximately 0 for the considered fhlA concentrations.

- [4]D. P. Bartel. MicroRNAs: genomics, biogenesis, mechanism, and function. *Cell*, 116(2):281–97, 2004.
- [5]S. Bernhart, H. Tafer, U. Mückstein, C. Flamm, P. Stadler, and I. Hofacker. Partition function and base pairing probabilities of RNA heterodimers. *Algorithms Mol Biol*, 1:3, 2006.
- [6]S. Brantl. Antisense-RNA regulation and RNA interference. *Bioch. Biophys. Acta*, 1575(1-3):15–25, 2002.
- [7]A. Busch, A. S. Richter, and R. Backofen. IntaRNA: Efficient prediction of bacterial sRNA targets incorporating target site accessibility and seed regions. *Bioinformatics*, page btn544, 2008.
- [8]S. Cao and S. Chen. Predicting RNA pseudoknot folding thermodynamics. *Nucleic Acids Res.*, 34:2634–2652, 2006.
- [9]J. Diamond, D. Turner, and D. Mathews. Thermodynamics of three-way multibranch loops in RNA. *Biochemistry*, 40:6971–6981, Jun 2001.
- [10]R. A. Dimitrov and M. Zuker. Prediction of hybridization and melting for double-stranded nucleic acids. *Biophysical Journal*, 87:215–226, 2004.
- [11]R. M. Dirks, J. S. Bois, J. M. Schaeffer, E. Winfree, and N. A. Pierce. Thermodynamic analysis of interacting nucleic acid strands. *SIAM Review*, 49(1):65–88, 2007.
- [12]R. M. Dirks and N. A. Pierce. A partition function algorithm for nucleic acid secondary structure including pseudoknots. *Journal of Computational Chemistry*, 24(13):1664–1677, 2003.
- [13]S. Gottesman. Micros for microbes: non-coding regulatory RNAs in bacteria. *Trends in Genetics*, 21(7):399–404, 2005.
- [14]G. J. Hannon. RNA interference. *Nature*, 418(6894):244–51, 2002.
- [15]T. Hjalte and E. Wagner. Bulged-out nucleotides in an antisense RNA are required for rapid target RNA binding in vitro and inhibition in vivo. *Nucleic Acids Res.*, 23:580–587, Feb 1995.
- [16]Y. Kato, T. Akutsu, and H. Seki. A grammatical approach to RNA-RNA interaction prediction. *Pattern Recognition*, 42(4):531–538, 2009.
- [17]L. D. Landau and E. M. Lifshitz. *Statistical Physics*. Pergamon, Oxford, UK, 1969.
- [18]N. Markham and M. Zuker. UNAFold: software for nucleic acid folding and hybridization. *Methods Mol. Biol.*, 453:3–31, 2008.
- [19]D. Mathews. Using an RNA secondary structure partition function to determine confidence in base pairs predicted by free energy minimization. *RNA*, 10:1178–1190, Aug 2004.
- [20]D. Mathews, J. Sabina, M. Zuker, and D. Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J. Mol. Biol.*, 288:911–940, May 1999.
- [21]D. Mathews and D. Turner. Experimentally derived nearest-neighbor parameters for the stability of RNA three- and four-way multibranch loops. *Biochemistry*, 41:869–880, Jan 2002.
- [22]J. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29:1105–1119, 1990.
- [23]U. Mückstein, H. Tafer, J. Hackermüller, S. Bernhart, P. Stadler, and I. Hofacker. Thermodynamics of RNA-RNA binding. *Bioinformatics*, 22:1177–1182, May 2006.
- [24]R. Nussinov, G. Piecznik, J. R. Grigg, and D. J. Kleitman. Algorithms for loop matchings. *SIAM Journal on Applied Mathematics*, 35:68–82, 1978.
- [25]D. Pervouchine. IRIS: intermolecular RNA interaction search. *Genome Inform*, 15:92–101, 2004.
- [26]J. Puglisi and I. Tinoco. Absorbance melting curves of RNA. *Meth. Enzymol.*, 180:304–325, 1989.
- [27]M. Rehmsmeier, P. Steffen, M. Hochsmann, and R. Giegerich. Fast and effective prediction of microRNA/target duplexes. *RNA*, 10:1507–1517, Oct 2004.
- [28]E. Rivas and S. Eddy. A dynamic programming algorithm for RNA structure prediction including pseudoknots. *J. Mol. Biol.*, 285:2053–2068, Feb 1999.
- [29]N. Seeman. From genes to machines: DNA nanomechanical devices. *Trends Biochem. Sci.*, 30:119–125, Mar 2005.
- [30]N. C. Seeman and P. S. Lukeman. Nucleic acid nanostructures: bottom-up control of geometry on the nanoscale. *Reports on Progress in Physics*, 68:237–270, Jan. 2005.
- [31]F. Simmel and W. Dittmer. DNA nanodevices. *Small*, 1:284–299, Mar 2005.
- [32]G. Storz. An expanding universe of noncoding RNAs. *Science*, 296(5571):1260–3, 2002.
- [33]S. Venkataraman, R. Dirks, P. Rothmund, E. Winfree, and N. Pierce. An autonomous polymerization motor powered by DNA hybridization. *Nat Nanotechnol*, 2:490–494, Aug 2007.
- [34]E. Wagner and K. Flardh. Antisense RNAs everywhere? *Trends Genet.*, 18:223–226, May 2002.
- [35]S. Walton, G. Stephanopoulos, M. Yarmush, and C. Roth. Thermodynamic and kinetic characterization of antisense oligodeoxynucleotide binding to a structured mRNA. *Biophys. J.*, 82:366–377, Jan 2002.
- [36]M. S. Waterman and T. F. Smith. RNA secondary structure: A complete mathematical analysis. *Math. Biosc.*, 42:257–266, 1978.
- [37]T. Xia, J. SantaLucia, M. Burkard, R. Kierzek, S. Schroeder, X. Jiao, C. Cox, and D. Turner. Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with Watson-Crick base pairs. *Biochemistry*, 37:14719–14735, Oct 1998.
- [38]P. Yin, R. Hariadi, S. Sahu, H. Choi, S. Park, T. Labean, and J. Reif. Programming DNA tube circumferences. *Science*, 321:824–826, Aug 2008.
- [39]P. D. Zamore and B. Haley. Ribo-gnome: the big world of small RNAs. *Science*, 309(5740):1519–24, 2005.
- [40]M. Zuker and P. Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Research*, 9(1):133–148, 1981.