DISTRIBUTION OF GRAPH-DISTANCES IN BOLTZMANN ENSEMBLES OF RNA SECONDARY STRUCTURES

Rolf Backofen[1−2], Markus Fricke[3], Manja Marz[3],
Jing Qin[4], and Peter F. Stadler[4−8]

[1] Department of Computer Science,Chair for Bioinformatics, University of Freiburg, Georges-Koehler-Allee 106, D-79110 Freiburg,
[2] Center for Biological Signaling Studies (BIOSS), Albert-Ludwigs-Universität, Freiburg, Germany
[3] Bioinformatics/High Throughput Analysis Faculty of Mathematics und Computer Science Friedrich-Schiller-University Jena Leutragraben 1, 07743 Jena
[4] Max Planck Institute for Mathematics in the Sciences, Inselstraße 22, 04103 Leipzig, Germany
[5] Bioinformatics Group, Department of Computer Science, and Interdisciplinary Center for Bioinformatics, University of Leipzig, Härtelstrasse 16-18, 04107 Leipzig, Germany
[6] Fraunhofer Institut for Cell Therapy and Immunology, Perlickstraße 1,04103 Leipzig, Germany
[7] Institute for Theoretical Chemistry, University of Vienna, Währingerstrasse 17, A-1090 Vienna, Austria
[8] Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM87501, USA.

Abstract. Large RNA molecules often carry multiple functional domains whose spatial arrangement is an important determinant of their function. Pre-mRNA splicing, furthermore, relies on the spatial proximity of the splice junctions that can be separated by very long introns. Similar effects appear in the processing of RNA virus genomes. Albeit a crude measure, the distribution of spatial distances in thermodynamic equilibrium therefore provides useful information on the overall shape of the molecule can provide insights into the interplay of its functional domains. Spatial distance can be approximated by the graph-distance in RNA secondary structure. We show here that the equilibrium distribution of graph-distances between arbitrary nucleotides can be computed in polynomial time by means of dynamic programming. A naive implementation would yield recursions with a very high time complexity of $O(n^{11})$. Although we were able to reduce this to $O(n^6)$ for many practical applications a further reduction seems difficult. We conclude, therefore, that sampling approaches, which are much easier to implement, are also theoretically favorable for most real-life applications, in particular since these primarily concern long-range interactions in very large RNA molecules.

1 Introduction

The distances distribution within an RNA molecule is of interest in various contexts. Most directly, the question arises whether panhandle-like structures (in which 3' and 5' ends of long RNA molecules are placed in close proximity) are the rule or an exception. Panhandles have been reported in particular for many RNA virus genomes. Several studies [1–4] agree based on different models that the two ends of single-stranded RNA molecules are typically not far apart. On a more technical level, the problem to compute the partition function over RNA secondary structures with given end-to-end distance d, usually measured as the number of external bases (plus possibly the number of structural domains) arises for instance when predicting nucleic acid secondary structure in the presence of single-stranded binding proteins [5] or in models of RNA subjected to pulling forces (e.g. in atom force microscopy or export through a small pore) [6–8]. It also plays a role for the effect of loop energy parameters [9].

In contrast to the end-to-end distance, the graph-distance between two *arbitrarily* prescribed nucleotides in a larger RNA structure does not seem to have been studied in any detail. However, this is of particular interest in the analysis of single-molecule fluorescence resonance energy transfer (smFRET) experiments [10]. This technique measures the extent of non-radiative energy transfer between donor and acceptor positions that are labeled by fluorescent dies. This efficiency of energy transfer, E_{FRET}, strongly depends on the spatial distance R according to $E_{FRET} = (1 + (R/R_0)^6)^{-1}$. The Förster radius R_0 sets the length scale, e.g. $R_0 \approx 54$ Å for Cy3-Cy5. FRET can measure spatial distance changes on the $20 - 100$ Å scale, compared to 3.4 Å for two stacked pairs.

It would be desirable in this context to compute the distribution of spatial distances for an equilibrium ensemble of 3D structures. Since this is not feasible in practice despite major progress in the field of RNA 3D structure prediction [11], we can only resort to considering the graph-distances on the ensemble of RNA secondary structures instead. While a crude approximation of reality, we shall see here that problem can be solved exactly using a dynamic programming approach.

2 Theory

2.1 RNA Secondary Structures

An RNA secondary structure is a vertex labeled outerplanar graph $G(V, \xi, E)$, where $V = \{1, 2, \ldots, n\}$ is a finite *ordered* set (of nucleotide positions) and $\xi : \{1, 2, \ldots, n\} \to \{\mathsf{A}, \mathsf{U}, \mathsf{G}, \mathsf{C}\}, i \mapsto \xi_i$ assigns to each vertex at position i (along the RNA sequence from 5' to 3') the corresponding nucleotide ξ_i. We write $\xi = \xi_1 \ldots \xi_n$ for the *sequence* underlying secondary structure and use $\xi[i \ldots j] = \xi_i \ldots \xi_j$ to denote the *subsequence* from i to j. The edge set E is subdivided into backbone edges of the form $\{i, i+1\}$ for $1 \leq i < n$ and a set B of base pairs satisfying the following conditions:

1. If $\{i,j\} \in B$ then $\xi_i\xi_k \in \{\mathsf{GC},\mathsf{CG},\mathsf{AU},\mathsf{UA},\mathsf{GU},\mathsf{UG}\}$.
2. If $\{i,j\} \in B$ then $|j-i| > 3$.
3. If $\{i,j\},\{i,k\} \in B$ then $j = k$
4. If $\{i,j\},\{k,l\} \in B$ and $i < k < j$ then $i < l < j$.

The first condition allows base pairs only for Watson-Crick and GU base pairs. The second condition implements the minimal steric requirement for an RNA to bend back on itself. The third condition enforces that B forms a matching in the secondary structure. The last condition (nesting condition) forbids crossing base pairs, i.e. pseudoknots.

The nesting condition results in a natural partial order in the set of base pairs B defined as $\{i,j\} \prec \{k,l\}$ if $k < i < j < l$. In particular, given an arbitrary vertex k, the set $B_k = \{\{i,j\} \in B | i \le k \le j\}$ of base pairs enclosing k is totally ordered. Note that k is explicitly allowed to be incident to its enclosing base pairs. A vertex k is *external* if $B_k = \emptyset$. A base pair $\{k,l\}$ is *external* if $B_k = B_l = \{\{k,l\}\}$.

Consider a fixed secondary structure G, for a given base pair $\{i,j\} \in B$, we say a vertex k is *accessible* from $\{i,j\}$ if $i < k < j$ and there is no other pair $\{i',j'\} \in B$ such that $i < i' < k < j' < j$. The unique subgraph $\mathcal{L}_{i,j}$ induced by i, j, and all the vertices accessible from $\{i,j\}$ is known as the *loop* of $\{i,j\}$. The *type* of a loop $\mathcal{L}_{i,j}$ is unique determined depending on whether $\{i,j\}$ is external or not, and the numbers of unpaired vertices and base pairs. For details, see [12]. Each secondary structure G has a unique set of loops $\{\mathcal{L}_{i,j} | \{i,j\} \in B\}$, which is called the *loop decomposition* of G. The free energy $f(G)$ of a given secondary structure, according to the standard energy model [13], is defined as the sum of the energies of all loops in its unique loop decomposition.

The relative location of two vertices v and w in G is determined by the base pairs B_v and B_w that enclose them. If $B_v \cap B_w \neq \emptyset$, there is a unique $\prec$-minimal base pair $\{i_{v,w}, j_{v,w}\}$ that encloses both vertices and thus a uniquely defined loop $\mathcal{L}_{\{i_{v,w},j_{v,w}\}}$ in the loop associated with v and w. If $B_v \setminus B_w = \emptyset$ or $B_w \setminus B_v = \emptyset$ then v or w is unpaired and part of $\mathcal{L}_{\{i_{v,w},j_{v,w}\}}$. Otherwise, i.e. $B_v \cap B_w = \emptyset$, there are uniquely defined $\prec$-maximal base pairs $\{k_v,l_v\} \in B_v \setminus B_w$ and $\{k_w,l_w\} \in B_w \setminus B_v$ that enclose v and w, respectively. This simple partition holds the key to computing distance distinguished partition functions below.

It will be convenient in the following to introduce edge weights $\omega_{i,j} = a$ if $j = i+1$, i.e., for backbone edges, and $\omega_{i,j} = b$ for $\{i,j\} \in B$. Given a path p, we define the weight of the path $d(p)$ as the sum of the weights of edges in the path. The (weighted) *graph-distance* $d^G_{v,w}$ in G is defined as the weight of the path p connecting v and w with $d(p)$ being minimal. For the weights, we require the following condition:

(W) If i and j are connected by an edge, then $\{i,j\} \in E$ is the unique shortest path between i and j.

This condition ensures that single edges cannot be replaced by detours of shorter weight. Condition (W) and property (ii) of the secondary structure graphs implies $b < 3a$ because the closing base pair must be shorter than a hairpin loop.

Furthermore, considering a stacked pair we need $b < b + 2a$, i.e. $a > 0$. We allow the degenerate case $b = 0$ that neglects the traversals of base pairs.

2.2 Boltzmann Distribution of Graph-Distances

For a fixed structure G, $d^G_{v,w}$ is easy to compute. Here, we are interested in the distribution $Pr[d^G_{v,w}|\xi]$ and its expected value $d_{v,w} = E[d^G_{v,w}|\xi]$ over the ensemble of all possible structures G for a given sequence ξ. Both quantities can be calculated from the Boltzmann distribution $Pr[G|\xi] = e^{-f(G)/RT}/Q$ where $Q = \sum_G e^{-f(G)/RT}$ denotes the partition function of the ensemble of structures. As first shown in [14], Q and related quantities can be computed in cubic time, see Appendix A. A crucial quantity for out task is the restricted partition function

$$Z^{v,w}[d] = \sum_{G \text{ with } d^G_{v,w}=d} e^{-f(G)/RT}$$

for a given pair v, w of positions in a given RNA sequence ξ. A simple but tedious computation (Appendix B) verifies that the $Pr[d^G_{v,w} = d|\xi] = Z^{v,w}[d]/Q$ and $d_{v,w} = E[d^G_{v,w}|\xi] = \sum_d (Z^{v,w}[d]/Q)d$. Hence it suffices to compute $Z^{v,w}[d]$ for $d = 1, \ldots, n$. In sections 2.3-2.5 we show that this can be achieved by a variant of McCaskill's approach [14].

For the ease of presentation we describe in the following only the recursion for the simplified energy model for the "circular maximum matching" matching, in which energy contributions are associated with individual base pairs rather than loops. Our approach easily extends to to the full model by using separating the partition functions into distinct cases for the loop types. We use the letter Z to denote partition functions with distance constraints, while Q is used for quantities that appear in McCaskill's algorithm and are considered as pre-computed here; they are defined more in detail in the Appendix A.

Before we continue with the calculation of the partition function, let's first look into problem formulation in more detail. For the FRET application, it is well-known that the rate which with FRET occurs is correlated with the distance. Therefore, only a limited range of distance changes (e.g. $20\mathring{A} - 100\mathring{A}$ for Cy3-Cy5) can be reported by the FRET experiments. Thus the more useful formulation of our problem is not to use the full expected quantity for all positions. Instead, we are interested in the average for all distances within some threshold θ_d. As the space and time complexity will depend on the number of distances we consider, we will parametrise our complexity by the number of nucleotides n and the number of overall distances considered $D = \theta_d + 1$, as well.

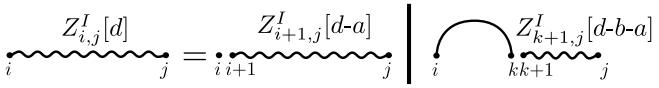
2.3 Recursions of $Z^{v,w}[d]$: v and w Are External

An important special case assumes that both v and w are external. This is case e.g. when v and w are bound by proteins. In particular, the problem of computing end-to-end distances, i.e., $v = 1$ and $w = n$, is of this type. Assuming (W), the shortest path between two external vertices v, w consists of the external vertices

and their backbone connections together with the external base pairs. We call this path the *inside path* of i, j since it does not involve any vertices "outside" the subsequence $\xi[i..j]$.

For efficiently calculating the internal distance between any two vertices v, w, we denote by $Z^I_{i,j}[d]$ the partition function over all secondary structures on $\xi[i..j]$ with end-to-end distance exactly d. Furthermore, let $Q^B_{i,j}$ denote the partition function over all secondary structures on $\xi[i..j]$ that are enclosed by the base pair $\{i, j\}$ (see Appendix A). We will later also need the partition function $Q_{i,j}$ over the sub-sequence $\xi[i..j]$, regardless of whether $\{i, j\}$ is paired or not.

Now note that any structure on $\xi[i..j]$ starts either with an unpaired base or with a base pair connecting i to some position k satisfying $i < k \leq j$. In the first case, we have $d^G_{i,j} = d^G_{i,i+1} + d^G_{i+1,j}$ where $d^G_{i,i+1} = a$. In the second case, there exists $d^G_{i,j} = d^G_{i,k} + d^G_{k,k+1} + d^G_{i+1,j}$ with $d^G_{i,k} = b$ and $d^G_{k,k+1} = a$. Thus, $Z^I_{i,j}[d]$ can be split as follows,

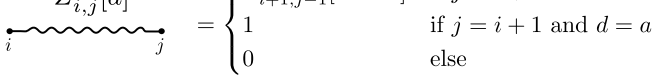


This gives the recursion

$$Z^I_{i,j}[d] = Z^I_{i+1,j}[d-a] + \sum_{i<k\leq j} Q^B_{i,k} Z^I_{k+1,j}[d-b-a] \tag{1}$$

with the initialization $Z^I_{ii}[0] = 1$ and $Z^I_{ii}[d] = 0$ for $d > 0$. For consecutive vertices we have $Z^I_{i,i+1}[a] = 1$ and $Z^I_{i,i+1}[d] = 0$ for $d \neq a$. These recursions have been derived in several different contexts, e.g. force induced RNA denaturations [6], the investigate of loop entropy dependence [9], the analysis of FRET signals in the presence of single-stranded binding proteins [5], as well as in mathematical studies of RNA panhandle-like structures [3, 4].

In the following it will be convenient to define also a special terms for the empty structure. Setting $Z^I_{i,i-1}[-a] = 1$ and $Z^I_{i,i-1}[d] = 0$ for $d \neq -a$ allows us to formally write an individual backbone edge as two edges flanking the empty structure and hence to avoid the explicit treatment of special cases. This definition of Z^I also includes the case that i and j are base paired in the recursion (1). This is covered by the case $k = j$, where we evaluate $Z^I_{j+1,j}[d-b-a]$. Since $d = b$ is the only admissible value here, this refers to $Z^I_{j+1,j}[-a]$, which has the correct value of 1 due to our definition. Later on, we will also need Z^I under the additional condition that the path starts and end with a backbone edge. We therefore introduce $Z^{I'}$ defined as defined as



By our initialization of Z^I, we can simply define $Z^{I'}$ by

$$Z^{I'}_{i,j}[d] = Z^I_{i+1,j-1}[d-2a] \tag{2}$$

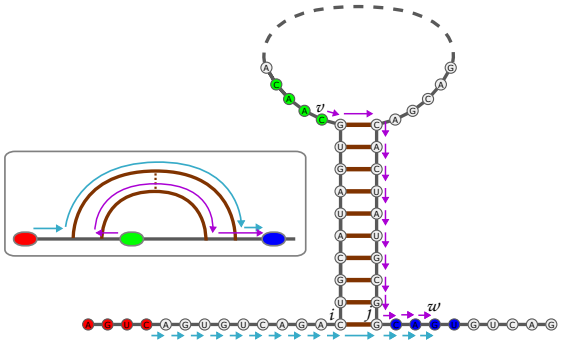


Fig. 1. Inside and outside paths. The shortest path (violet arrows) from v (green) to w (blue) is not an inside path: *inside* emphasizes that, in contrast to the shortest path (cyan arrows) between the red region and v, it is not contained in the interval determined by its end points.

Note that if $Z^{I'}_{i,j}[d]$ is called with $j = i+1$, then we call $Z^{I}_{i+1,i}[d-2a]$. The only admissible value again is the correct value $d = a$.

This recursion requires $O(Dn^3)$ time and space. It is possible to reduce the complexity in this special case by a linear factor. The trick is to use conditional probabilities for arcs starting at i or the conditional probability for i to be single-stranded, which can be determined from the partition function for RNA folding [3], see Appendix C.

2.4 Recursions of $Z^{v,w}[d]$: The General Case

The minimal distance between two positions that are covered by an arc can be realized by *inside paths* and *outside paths*. This complicates the algorithmic approach, since both types of paths must be controlled simultaneously. Consider Fig. 1. The shortest path between the green and blue regions includes some vertices outside the interval between these two regions. The basic idea is to generalize Equation (1) to computing the partition function $Z^{v,w}[d]$. The main question now becomes how to recurse over decompositions of both the inside and the outside paths.

Fig. 1 shows that the outside paths are important for the green region, i.e., the region that is covered by an arc. Hence, we have to consider the different cases that the two positions v and w are covered by arcs. The set Ω of all secondary structures on ξ can be divided into two disjoint subclasses that have to be treated differently:

Ω_0 v and w are not enclosed in a common base pair, i.e., $B_v \cap B_w = \emptyset$.
Ω_1 there is a base pair enclosing both v and w, i.e., $B_v \cap B_w \neq \emptyset$.

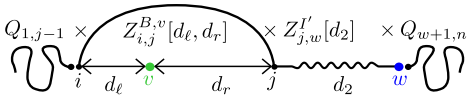
Note that this bipartition explicitly depends on v and w. In the following, we will first introduce the recursions that are required in Ω_0 structures to compute $Z^{v,w}[d]$.

Contribution of Ω_0 structures to $Z^{v,w}[d]$ One example of this case is given in Fig. 1 with the red and blue region, where v (vertex in green region) is covered by an arc, and w (vertex in blue region) is external. Denote the $\prec$-maximal base

pair enclosing v by $\{i, j\}$. Since at most one of v and w is covered by an arc, we know that $j < w$. Hence, every path p from v to w, and hence also the shortest paths (not necessarily unique) must run through the right end j of the arc $\{i, j\}$. More precisely, there must sub-paths p_1 and p_2 with $d(p) = d(p_1) + d(p_2) + a$ such that $v \overset{p}{\rightsquigarrow} w \rightarrow v \overset{p_1}{\rightsquigarrow} j - (j+1) \overset{p_2}{\rightsquigarrow} w$, where $i \overset{p}{\rightsquigarrow} j$ denotes that p is **a** shortest path from i to j and $-$ denotes a single backbone edge. For the shortest path from v to j, it consists either of a shortest path $v \overset{p'}{\rightsquigarrow} i$ and the arc $\{i, j\}$, or it goes directly to j without using the arc $\{i, j\}$.

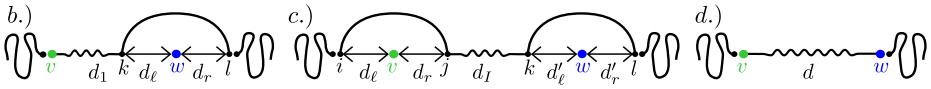
How does this distinction translate to the partition function approach? If we want to calculate the contribution of this case to the partition function $Z^{v,w}[d]$, we have to split both the sequence $\xi[i, w]$ and distance d as follows

a.)



where $Z^{I'}_{j,w}[d_2]$ is the partition function starting and ending with a single-stranded base as defined in Equation (2), and $Z^{B,v}_{i,j}[d_\ell, d_r]$ is the partition function consisting of all structures of $\xi[i, j]$ containing the base pair $\{i, j\}$ with the property that the shortest path from v to i has length d_ℓ and the shortest path from v to j has length d_r. In addition, d, d_r and d_2 must satisfy $d = d_r + d_2$.

The remaining cases for the contribution of the class Ω_0 to $Z^{v,w}[d]$ are given by all other possible combinations of v and w being single-stranded or being covered by an arc, i.e.,



To simplify, we extend the definition of $Z^{B,v}_{i,j}[d_\ell, d_r]$ by setting $Z^{B,v}_{v,v}[0, 0] = 1$ and $Z^{B,v}_{v,v}[d_\ell, d_r] = 0$ for $d_\ell + d_r > 0$. This allows us to conveniently model all cases where either v or w are external, i.e., a.), b.), and d.), as special cases of c.).

In case c.) we have to split the distance d into four contributions and we require two splitting positions for the sequence for all combinations of i, j, v, w. This would result in an $O(n^6 D^5)$ algorithm. A careful inspection shows, however, that the split of the distances for the arcs into d_ℓ and d_r is unnecessary. Since we want to know only distance to the left/right end overall, we can simply introduce two matrices $Z^{B,v,\ell}_{i,j}[d]$ and $Z^{B,v,r}_{i,j}[d]$ that store these values. These matrices can be generated from $Z^{B,v}_{i,j}[d_\ell, d_r]$ as follows:

$$Z^{B,v,\ell}_{i,j}[d] = \sum_{\substack{d_r \\ d_r + b \geq d}} Z^{B,v}_{i,j}[d, d_r] + \sum_{\substack{d_\ell \\ d_\ell > d}} Z^{B,v}_{i,j}[d_\ell, d - b]$$

Analogously, we compute $Z^{B,v,r}_{i,j}[d]$.

Overall, the contribution to $Z^{v,w}[d]$ for structures in Ω^0 is given by

$$Z_0^{v,w}[d] = \sum_{\substack{d_1,d_2 \\ d_1+d_2 \le d}} \sum_{\substack{i,j,k,l \\ i \le v \le j < k \le w \le l}} \begin{pmatrix} Q_{1,i-1} \cdot Z_{i,j}^{B,v,r}[d_1] \\ \cdot Z_{j,k}^{I'}[d-(d_1+d_2)] \\ \cdot Z_{k,l}^{B,w,\ell}[d_2] \cdot Q_{l+1,n} \end{pmatrix} \tag{3}$$

Note that for splitting the distance, we reuse the same indices (e.g., the j in $Z_{i,j}^{B,v,r}[d_1] \cdot Z_{j,k}^{I'}[d-(d_1+d_2)]$, where as for the remaining partition function, we use successive indices (e.g.,the i in $Q_{1,i-1} \cdot Z_{i,j}^{B,v,r}[d_1]$). This difference comes from the fact that splitting a sequence into subsequences is done naturally between two successive indices, whereas splitting a distance is naturally done by splitting at an individual position. We have only to guarantee that the substructures which participate in the split do agree on the structural context of the split position. This is guaranteed by requiring that $Z^{I'}$ starts and ends with a backbone edge. We note that the incorporation of the full dangling end parameters makes is more tedious to handle the splitting positions.

This results in a complexity of $O(n^6D^3)$ time and $O(n^3D)$ space. However, we do not need to split in i, j, k, j simultaneously. Instead, we could split case (c) at position j and introduce for all $v \le j$ and $k \le w$ the auxiliary variables

$$Z_{1,j}^{B,v,r}[d_1] = \sum_{i \le v} Q_{1,i-1} \cdot Z_{i,j}^{B,v,r}[d_1] \qquad Z_{k,n}^{B,w,\ell}[d_2] = \sum_{w \le l} Z_{k,l}^{B,w,\ell}[d_2] \cdot Q_{l+1,n}$$

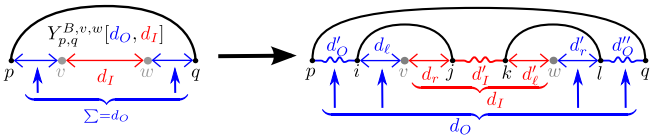
$$Z_{j,n}^{IB,w,\ell}[d'] = \sum_{k>j} \sum_{d_2 \le d'} Z_{j,k}^{I'}[d'-d_2] \cdot Z_{k,n}^{B,w,\ell}[d_2].$$

Finally, we can replace recursion (3) by

$$Z_0^{v,w}[d] = \sum_{v \le j} \sum_{d_1 \le d} Z_{1,j}^{B,v,r}[d_1] \cdot Z_{j,n}^{IB,w,\ell}[d-d_1] \tag{4}$$

We thus arrive at $O(n^4D^2)$ time and $O(n^3D)$ space complexity for the contribution of Ω_0 structures to $Z^{v,w}[d]$, excluding the complexity of computing $Z_{i,j}^{B,v}[d_\ell, d_r]$.

Contribution of Ω_1 structures to $Z^{v,w}[d]$ Ω_1 contains all cases where v and w are covered by a base pair. In the following, let $\{p,q\}$ be the $\prec$-minimal base pair covering v and w. In principle, this case looks similar to the overall case for Ω_0. However, we have now to deal not only with an inside distance, but also with an outside distance over the base pair $\{p,q\}$. Thus, we need to store the partition function for all inside and outside for each $\prec$-minimal arc $\{p,q\}$ that covers v and w, which we will call $Y_{p,q}^{B,v,w}[d_O, d_I]$. Now in principle, we a similar recursion as defined for Z_0 in equation (3), with the additional complication that we have to take care of the additional outside distance due to the arc $\{p,q\}$. Thus, we obtain the following splitting:



Again we can avoid the complexity of simultaneously splitting at $\{i,j\}$ *and* $\{k,l\}$ by doing a major split after j. Thus, we get the equivalent recursions as in eqns.(5–7):

$$Y^{B,v,r}_{p,j}[d,d_r] = \sum_{p<i\le v}\sum_{d'_O\le d} Z^{I'}_{p,i}[d'_O]\cdot Z^{B,v}_{i,j}[\overbrace{d-d'_O}^{\hat{=}\,d_\ell},d_r] \tag{5}$$

$$Y^{B,w,\ell}_{k,q}[d'_\ell,d] = \sum_{w\le l<q}\sum_{d''_O\le d} Z^{B,w}_{k,l}[d'_\ell,\overbrace{d-d'_O}^{\hat{=}\,d'_r}]\cdot Z^{I'}_{l,q}[d''_O] \tag{6}$$

$$Y^{IB,w,\ell}_{j,q}[d'_I,d] = \sum_{j<k<q}\sum_{d'_\ell\le d'_I} Z^{I'}_{j,k}[d'_I-d'_\ell]\cdot Y^{B,w,\ell}_{k,q}[d'_\ell,d] \tag{7}$$

Overall, we get the following recursion:

$$Z^{v,w}_{p,q}[d_O,d_I] = \sum_{v\le j}\sum_{\substack{d_r\le d_I\\ d\le d_O}} Y^{B,v,r}_{p,j}[d,d_r]\cdot Y^{IB,w,\ell}_{q,j}[d_I-d_r,d_O-d] \tag{8}$$

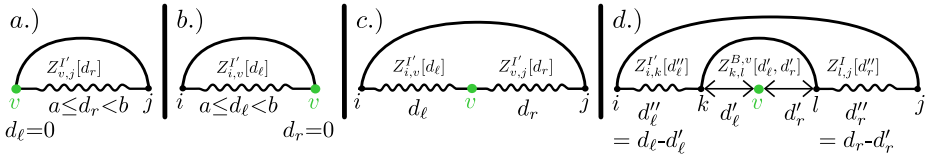
Overall, we can now define $Z^{v,w}[d]$ by

$$Z^{v,w}[d] = Z^{v,w}_0[d] + \sum_{\substack{\{p,q\}\neq\{v,w\}\\ d_I\ge d+b}} Z^{v,w}_{p,q}[d,d_I] + \sum_{\substack{\{p,q\}\neq\{v,w\}\\ d<d_O+b}} Z^{v,w}_{p,q}[d_O,d]$$

This part has now a complexity of $O(n^4D^2)$ space and $O(n^5D^4)$ time. For practical applications, however, we do not need to consider all possible $\{p,q\}$. Instead, there are only few base pairs that are likely to form *and* that cover v,w, especially for v,w where the internal distance of v,w is large enough such that an outside path has to be considered at all. If we assume a constant number of such long-range base-pairs, then the complexity is reduced by an n^2-factor. For the complexity in terms of distance, recall that D is typically small.

2.5 Recursions for $Z^{B,v}_{i,j}[d_\ell, d_r]$

So far, we have used $Z^{B,v}_{i,j}[d_\ell,d_r]$ as a black box. In order to compute these terms, we distinguish the limiting cases a.) $v=i$, b.) $v=j$, c.) is external from the generic case d.):



Starting from the limiting cases, we initialize $Z^{B,v}_{v,j}[0,d_r]$ as follows:

$$Z^{B,v}_{v,j}[0,d_r] = \begin{cases} Z^{I'}_{v,j}[d_r] & \text{for } a \le d_r < b \\ \sum_{d' \ge b} Z^{I'}_{v,j}[d'] & \text{for } d_r = b \\ 0 & \text{otherwise} \end{cases}$$

and analogously for $Z^{B,v}_{i,v}[d_\ell,0]$. Furthermore, $Z^{B,v}_{i,j}[0,0] = 0$ for $i \neq v \neq j$. Finally, we have the following recursion for $i \neq v \neq j$, $d_\ell > 0$ and $d_r > 0$:

$$Z^{B,v}_{i,j}[d_\ell,d_r] = \widehat{Q}^b_{i,j} \cdot \sum_{\substack{k\neq l \\ i<k\le v \\ v\le l<j}} \sum_{\substack{d'_\ell \le d_\ell \\ d'_r \le d_r}} Z^{I'}_{i,k}[d_\ell - d'_\ell] \cdot Z^{B,v}_{k,l}[d'_\ell,d'_r] \cdot Z^{I'}_{l,j}[d_r - d'_r] \quad (9)$$

where $\widehat{Q}^b_{i,j}$ is the external partition function over all structures on the union of the intervals $\xi[1..i] \cup \xi[j..n]$ so that $\{i,j\}$ is a base pair. This is equivalent to $\widehat{Q}^b_{i,j} = Pr(\{i,j\}) \times Q/Q^b_{i,j}$. The base pair probability $Pr(\{i,j\})$, and the partition functions Q and $Q^b_{i,j}$ are computed by means of McCaskill's algorithm.

Recursion (9) apparently has complexity $O(n^5D^4)$ in time and $O(n^3D^2)$ in space. This can be reduced due to the strong dependency between d_ℓ and d_r, however. By construction we have $|d_\ell - d_r| \le b$ since we can always use the bond $\{i,j\}$ to traverse from one end to the other. Furthermore, assuming integer values for a and b, we can have only $c_b = 2b/\operatorname{lcd}(a,b) + 1$ different values for $(d_\ell - d_r)$ This implies that the space complexity of $Z^{B,v}_{i,j}[d_\ell,d_r]$ is $O(n^4c_b)$. Instead of $Z^{B,v}_{i,j}[d_\ell,d_r]$, we store $Z^{B,v}_{i,j}[d_\ell,d_\ell+d_{\text{add}}]$ for the c_b possible values of d_{add}.

The dependency between d_ℓ and d_r can also be used to reduce the time complexity in Equ.(9). The problematic case is (d). Instead of using the variables d_ℓ and d_r in $Z^{B,v}_{i,j}[d_\ell,d_r]$ we use the pair d_ℓ, d_{add} in $Z^{B,v}_{i,j}[d_\ell,d_\ell+d_{\text{add}}]$. Similarly, we use d'_ℓ, d'_{add} instead of d'_ℓ, d'_r for the inner base pair, which then determines completely the splitting the distances. The details are relegated to Appendix D. Overall, this results in an recursion for $Z^{B,v}_{i,j}[d_\ell,d_\ell+d_{\text{add}}]$ with complexity $O(n^5c_b^2)$ time and $O(n^3Dc_b)$ space.

3 Discussion and Applications

The theoretical analysis of the distance distribution problem shows that, while polynomial-time algorithms exist, they probably cannot the improved to space and time complexities that make them widely applicable to large RNA molecules. We therefore resort to sampling Boltzmann-weighted secondary structures with `RNAsubopt -p` [15] in which the dynamic programming routine is introduced in [16]. As the graph-distance for a pair of nucleotides can be computed in $O(n \log n)$ even large samples can be evaluated efficiently[9].

[9] The C++ program `RNAgraphdist` is available from http://www.rna.uni-jena.de/supplements/RNAgraphdist/RNAgraphdist1.0.tar.gz

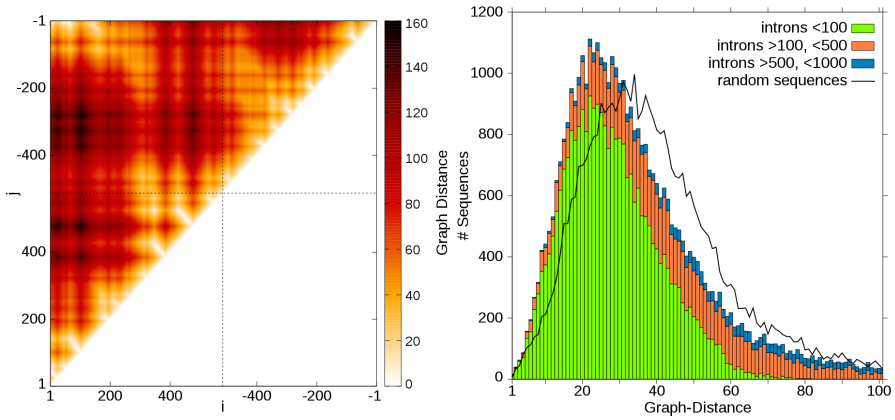


Fig. 2. Left: Distribution of graph-distances ($a = b = 1$) in *Drosophila melanogaster* pre-mRNAs between the first and last intron position. To save computational resources, pre-mRNAs were truncated to 100 nt flanking sequence. The black curve shows the graph-distance distribution computed for the corresponding pairs of positions on sequences that were randomized by di-nucleotide shuffling. Right: Graph-distances ($a = b = 1$) within and between the 5' and 3' regions of the genomic RNA of human *Coronavirus* 229E computed from a concatenation of position 1–576 and 25188–25688. Secondary structures bring the 5' TRS-L and 3' TRS-B elements into close proximity.

Long-range interactions play an important role in pre-mRNA splicing and in the regulation of alternative splicing [17, 18], bringing splice donor, acceptor, branching site into close spatial proximity. Fig. 2(Left) shows for *D. melanogaster* pre-mRNAs that the distribution of graph-distances between donor and acceptor sites shifted towards smaller values compared to randomly selected pairs of positions with the same distance. Although the effect is small, it leaves a clear statistical signal.

The spatial organization of the genomic and sub-genomic RNAs is important for the processing and functioning of many RNA viruses. This goes far beyond the well-known panhandle structures. In *Coronavirus* the interactions of the 5' TRS-L cis-acting element with body TRS elements has been proposed as an important determinant for the correct assembly of the *Coronavirus* genes in the host [19]. The matrix of expected graph-distances in Fig. 2(Right) shows that TRS-L and TRS-B are indeed placed near each other.

Our first results show that the systematic analysis of the graph-distance distribution both for individual RNAs and their aggregation over ensembles of structures can provide useful insights into structural influences on RNA function. These may not be obvious directly from the structures due to the inherent difficulties of predicting long-range base pairs with sufficient accuracy and the many issues inherent in comparing RNA structures of very disparate lengths.

Acknowledgements. This work was supported in part by the *Deutsche Forschungsgemeinschaft* proj. nos. BA 2168/2-2, STA 850/10-2, SPP 1596 and MA5082/1-1.

References

1. A. M. Yoffe, P. Prinsen, W. M. Gelbart, and A. Ben-Shaul. The ends of a large RNA molecule are necessarily close. *Nucl. Acids Res.*, 39:292–299, 2011.
2. L. T. Fang. The end-to-end distance of RNA as a randomly self-paired polymer. *J. Theor. Biol.*, 280:101–107, 2011.
3. P. Clote, Y. Ponty, and J. M. Steyaert. Expected distance between terminal nucleotides of RNA secondary structures. *J. Math. Biol.*, 65:581–599, 2012.
4. H. S. Han and C. M. Reidys. The 5'-3' distance of RNA secondary structures. *J. Comput. Biol.*, 19:867–878, 2012.
5. R. A. Forties and R. Bundschuh. Modeling the interplay of single-stranded binding proteins and nucleic acid secondary structure. *Bioinformatics*, 26:61–67, 2010.
6. U. Gerland, R. Bundschuh, and T. Hwa. Force-induced denaturation of RNA. *Biophys. J.*, 81:1324–1332, 2001.
7. M. Müller, F. Krzakala, and M. Mézard. The secondary structure of RNA under tension. *Eur. Phys. J. E*, 9:67–77, 2002.
8. U. Gerland, R. Bundschuh, and T. Hwa. Translocation of structured polynucleotides through nanopores. *Phys. Biol.*, 1:19–26, 2004.
9. T. R. Einert, P. Näger, H. Orland, and R. Netz. Impact of loop statistics on the thermodynamics of RNA folding. *Phys. Rev. Lett.*, 101:048103, 2008.
10. R. Roy, S. Hohng, and T. Ha. A practical guide to single-molecule FRET. *Nature Methods*, 5:507 – 516, 2008.
11. R. Das and D. Baker. Automated de novo prediction of native-like RNA tertiary structures. *Proc. Natl. Acad. Sci. USA*, 104:14664–14669, 2007.
12. P. Schuster, W. Fontana, P. F. Stadler, and I. L. Hofacker. From sequences to shapes and back: a case study in RNA secondary structures. *Proc. Royal Society London B*, 255(1344):279–84, 1994.
13. D. H. Mathews, M. D. Disney, J. L. Childs, S. J. Schroeder, M. Zuker, and D. H. Turner. Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proc. Natl. Acad. Sci. USA*, 101:7287–7292, 2004.
14. J. S. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29(6-7):1105–19, 1990.
15. R. Lorenz, S. H. Bernhart, C. Höner zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, and I. L. Hofacker. ViennaRNA Package 2.0. *Alg. Mol. Biol.*, 6:26, 2011.
16. Y. Ding and C.E. Lawrence. A statistical sampling algorithm for RNA secondary structure prediction. *Nucl. Acids Res.*, 31(24):7280–7301, 2003.
17. A. P. Baraniak, E. L. Lasda, E. J. Wagner, and M. A. Garcia-Blanco. A stem structure in fibroblast growth factor receptor 2 transcripts mediates cell-type-specific splicing by approximating intronic control elements. *Mol. Cell Biol.*, 23:9327–9337, 2003.
18. C. J. McManus and B. R. Graveley. RNA structure and the mechanisms of alternative splicing. *Curr. Opin. Genet. Dev.*, 21:373–379, 2011.
19. D. Dufour, P. A. Mateos-Gomez, L. Enjuanes, J. Gallego, and I. Sola. Structure and functional relevance of a transcription-regulating sequence involved in coronavirus discontinuous RNA synthesis. *J. Virol.*, 85(10):4963–4973, 2011.

Supplemental Material

The following **Supplemental Material** here is intended to provide some additional information and details that could not be included in the main text due to length restrictions.

We hope that it is useful for the review process. The material will be made available at the authors' web site in case the manuscript is accepted.

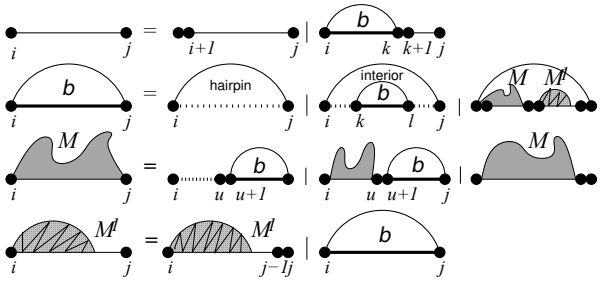


Fig. 1. Recursive decomposition of RNA secondary structures.

Appendix A: Proof of the $O(n^3)$ Complexity of McCaskill's Algorithm

Dynamic programming algorithms for secondary structure prediction are based on a simple recursive decomposition: any feasible structure on the interval $[i, j]$ has the first base either unpaired or paired with a position k satisfying $i < k \leq j$. The condition that base pairs do not cross implies that the intervals $[i+1, k-1]$ and $[k+1, j]$ form self-contained structures whose energies can be evaluated independent of each other. In conjunction with the standard energy model [1], which distinguishes hairpin loops, interior loops (including stacked base pairs), and multiloops, this leads to the recursions diagrammatically represented in Fig. 1. This algorithmic approach was pioneered e.g. in [2, 3] and is also used in the `ViennaRNA Package` [4].

Appendix B: Proof of the $E[d_G(v,w)] = \sum_d d \times \frac{Z^{v,w}[d]}{Z}$

Proof of $E[d_G(v,w)] = \sum_d d \times \frac{Z^{v,w}[d]}{Z}$.

$$E[d_G(v,w)] = \sum_G d_G(v,w) \times Pr[G|\xi] \;=\; \sum_d \sum_{G \text{ with } d_G(v,w)=d} d \times \frac{e^{-f(G)/RT}}{Z}$$
$$= \sum_d d \times \frac{\sum_{G \text{ with } d_G(v,w)=d} e^{-f(G)/RT}}{Z} \;= \sum_d d \times \frac{Z^{v,w}[d]}{Z}$$

Appendix C: Proof of Lemma 1

Lemma 1. *The expected distance* $E[d^G_{i,j}]$ *can be calculated as:*

$$E[d^G_{i,j}] = (a + E[d^G_{i+1,j}]) \cdot \frac{1 \cdot Q_{i+1,j}}{Q_{i,j}} + \sum_{i<k\leq j} (b + E[d^G_{k+1,j}]) \cdot \frac{Q^b_{i,k} \cdot Q_{k+1,j}}{Q_{i,j}} \quad (10)$$

Proof of Lemma 1.

Let G be a structure. For simplicity of notation, we write $G = \bullet G'$ if the first position is unpaired, and $G = (\ldots)_j G'$ is the first base is paired to some position j, and G' is the substructure of G starting from position $j+1$. Alternatively, we may use the notation $(i,j) \in G$ for the case where the position i and j are base paired in G.

The expected length $E[d_G(i,j)]$ can be calculated as: follows:

$$\begin{aligned} E[d_G(i,j)] &= \sum_{G \text{ struct. of } \xi[i\ldots j]} d_G(i,j) Pr[G|\xi[i\ldots j]] \\ &= \sum_{G=\bullet G'} (a + d_{G'}(i,j)) Pr[G|\xi[i\ldots j]] \quad + \sum_{i<k\leq j} \sum_{G=(\ldots)_k G'} (b + d_{G'}(i,j)) Pr[G|\xi[i\ldots j]] \\ &\overset{def.}{=} \qquad EL_{sg} \qquad\qquad\qquad\qquad + \sum_{i<k<j} EL_{bp(k)} \end{aligned}$$

Now EL_{sg} can be simplified as follows:

$$\begin{aligned} EL_{sg} &= \sum_{G=\bullet G'} (a + d_{G'}(i,j)) Pr[G|\xi[i\ldots j]] \\ &= \left(\sum_{G=\bullet G'} a \cdot Pr[G|\xi[i\ldots j]] \right) + \left(\sum_{G=\bullet G'} d_{G'}(i,j) \cdot Pr[G|\xi[i\ldots j]] \right) \\ &= a \cdot Pr[G = \bullet G'|\xi[i\ldots j]] + \left(\sum_{G=\bullet G'} d_{G'}(i,j) \cdot Pr[G|\xi[i\ldots j]] \right), \end{aligned}$$

where $Pr[G = \bullet G' | \xi[i \dots j]]$ can be calculated as the probability of the first position to be single-stranded in the sequence $\xi[i \dots j]$, i.e.,

$$Pr[G = \bullet G' | \xi[i \dots j]] = \frac{1 \cdot Q_{i+1,j}}{Q_{i,j}}$$

We are also able to push the second term since

$$\sum_{G = \bullet G'} d_{G'}(i,j) \cdot Pr[G | \xi[i \dots j]] = \sum_{G'} d_{G'}(i,j) \cdot Pr[\bullet G' | \xi[i \dots j]]$$

Now we know that for every G' we have that the Boltzmann weighted energy of G' is part of the partition function of $Q_{i+1,j}$. Thus we get

$$\begin{aligned}
&= \sum_{G'} d_{G'}(i,j) \cdot \frac{\exp(-E(\bullet G')/kT)}{Q_{i,j}} \\
&= \sum_{G'} d_{G'}(i,j) \cdot \frac{\exp(-E(G')/kT)}{Q_{i+1,j}} \frac{Q_{i+1,j}}{Q_{i,j}} \\
&= \frac{Q_{i+1,j}}{Q_{i,j}} \sum_{G'} d_{G'}(i,j) \cdot \frac{\exp(-E(G')/kT)}{Q_{i+1,j}} \\
&= Pr[G = \bullet G' | \xi[i \dots j]] \sum_{G'} d_{G'}(i,j) \cdot Pr[G' | \xi[i+1 \dots j]] \\
&= Pr[G = \bullet G' | \xi[i \dots j]] \cdot E[d_G(i+1,j)]
\end{aligned}$$

Overall we get

$$EL_{sg} = (a + E[d_G(i+1,j)]) \cdot Pr[G = \bullet G' | \xi[i \dots j]]$$

For the term $EL_{bp(k)}$, we have a similar reduction:

$$\begin{aligned}
EL_{bp(k)} &= \sum_{G = (\dots)_k G'} (b + d_{G'}(i,j)) Pr[G | \xi[i \dots j]] \\
&= \left(\sum_{G = (\dots)_k G'} b \cdot Pr[G | \xi[i \dots j]] \right) + \left(\sum_{G = (\dots)_k G'} d_{G'}(i,j) Pr[G | \xi[i \dots j]] \right) \\
&= (b \cdot Pr[G = (\dots)_k G' | \xi[i \dots j]]) + \left(\sum_{G = (\dots)_k G'} d_{G'}(i,j) Pr[G | \xi[i \dots j]] \right),
\end{aligned}$$

where $Pr[G = (\dots)_k G' | \xi[i \dots j]] = \frac{Q^b_{ik} \cdot Q_{k+1,j}}{Q_{i,j}}$.

Now

$$\begin{aligned}
\sum_{G=(\ldots)_k G'} d_{G'}(i,j) Pr[G|\xi[i\ldots j]] &= \sum_{G'} \sum_{G''=(G''')_k} d_{G'}(i,j) Pr[G''G'|\xi[i\ldots j]] \\
&= \sum_{G'} \sum_{G''=(G''')_k} d_{G'}(i,j) \frac{\exp(E(G'')/kT)\exp(G'/kT)}{Q_{ij}} \\
&= \sum_{G'} \sum_{G''=(G''')_k} d_{G'}(i,j) \frac{\exp(E(G'')/kT)\exp(G'/kT)}{Q_{ij}} \\
&= \sum_{G'} d_{G'}(i,j) \frac{\left(\sum_{G''=(G''')_k} \exp(E(G'')/kT)\right)\exp(G'/kT)}{Q_{ij}} \\
&= \sum_{G'} d_{G'}(i,j) \frac{Q^b_{i,k}\exp(G'/kT)}{Q_{ij}}
\end{aligned}$$

Now we can again simply extend by $Q_{k+1,j}$, getting

$$\begin{aligned}
&= \sum_{G'} d_{G'}(i,j) \frac{Q^b_{i,k} \cdot Q_{k+1,j} \cdot \exp(G'/kT)}{Q_{ij} \cdot Q_{k+1,j}} \\
&= \sum_{G'} d_{G'}(i,j) \frac{Q^b_{i,k} \cdot Q_{k+1,j}}{Q_{ij}} \cdot \frac{\exp(G'/kT)}{Q_{k+1,j}} \\
&= Pr[G = (\ldots)_k G'|\xi[i\ldots j]] \sum_{G'} d_{G'}(i,j) Pr[G'|\xi[k+1\ldots j]] \\
&= Pr[G = (\ldots)_k G'|\xi[i\ldots j]] \cdot E[d_G(k+1,j)]
\end{aligned}$$

Overall we get

$$EL_{bp(k)} = (b + E[d_G(k+1,j)]) \cdot Pr[G = (\ldots)_k G'|\xi[i\ldots j]]$$

and thus the second summand.

Appendix D: Recurrsions of different cases for $Z^{B,v}_{i,j}$.

Now there are three sub-cases (see Figure 2). If $-b < d_{add} < +b$, then we know that neither a shortest path $v \overset{p}{\rightsquigarrow} i$ nor $v \overset{p}{\rightsquigarrow} j$ uses the arc $\{i,j\}$. The left distance is thus given by $d_\ell - d'_\ell$. Using the shortcuts $d_r = d_\ell + d_{\text{add}}$ and $d'_r = d'_\ell + d'_{\text{add}}$, then the distance between l and j must be $d_r - d'_r = (d_\ell + d_{\text{add}}) - (d'_\ell + d'_{\text{add}})$. If, on the other hand, $d_{\text{add}} = +b$, then we know that there is at least one shortest path that can be composed by using a shortest path $v \rightsquigarrow i$, followed by the arc $\{i,j\}$. This of course implies that the shortest path $v \overset{p}{\rightsquigarrow} j$ is has exactly the length $d_\ell + b$, or is larger. For a sub-path $l+1 \overset{p'}{\rightsquigarrow} j$ this implies that the length is greater or equal $d = d_r - d'_r = (d_\ell + b) - (d'_\ell + d'_{\text{add}})$. Thus, we just have to

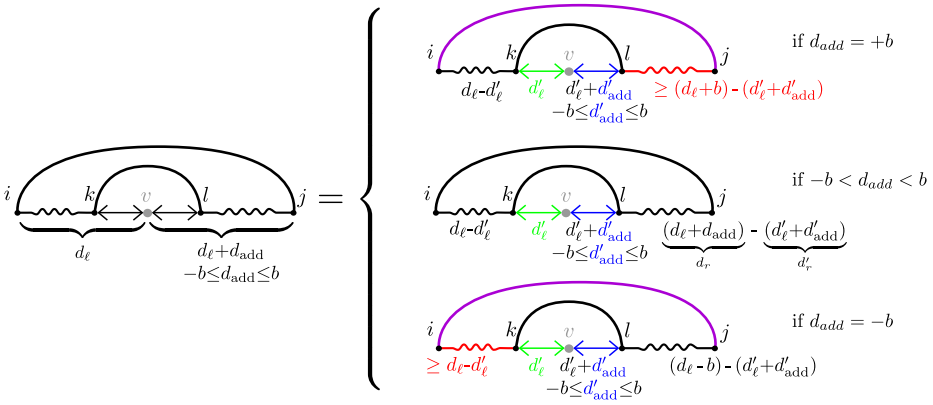


Fig. 2. Different cases for $Z^{B,v}_{i,j}[d_\ell, d_\ell + d_{\text{add}}]$. The values that are choose to split d_ℓ and d_{add} are indicated in green and blue. When the arc $\{i, j\}$ is colored violet, then there is a shortest path that does not use the distance marked in red but uses the other direction together with the arc (i, j).

add all partition functions $Z^{I'}_{k,j}[d']$ with $d' > d$. This can be done efficiently by using a precalculated matrix $Z^{I'\geq}_{i,j}[d]$, which is defined as $\sum_{d' \geq d} Z^{I'}_{i,j}[d']$. Note that $Z^{I'\geq}_{i,j}[d]$ can also be defined if we restrict in all recursion the distance d to a threshold θ_d, since $Z^{I'\geq}_{i,j}[d] = \sum_{d' \geq d} Z^{I'}_{i,j}[d'] = Q'_{i,j} - \sum_{d'<d} Z^{I'}_{i,j}[d']$. In which, where $Q'_{i,j}$ is $Q_{i+1,j-1}$ if $j > i + 1$, 1 if $j = i + 1$ and 0 otherwise. Note, furthermore, that all $Z^{I'}_{i,j}[d']$ for $d' < d \leq \theta_d$ are calculated when we restrict the distance to θ_d

Finally, if $d_{\text{add}} = -b$, then the shortest path $l \overset{p}{\rightsquigarrow} j$ has distance $(d_\ell - b) - (d'_\ell + d'_{\text{add}})$. For the shortest path $k \overset{p}{\rightsquigarrow} i$, we know that it has length $d_\ell - d'_\ell$ or greater, which can be resolved by again using $Z^{I'\geq}_{i,k-1}[d_\ell - d'_\ell]$. Overall, we get the following optimized recursion for $Z^{B,v}_{i,j}[d_\ell, d_\ell + d_{\text{add}}]$ with $d_\ell \neq 0$ and

$d_\ell + d_{\text{add}} \neq 0$:

$$Z^{B,v}_{i,j}[d_\ell, d_\ell + d_{\text{add}}] =$$

$$\widehat{Q}^b_{i,j} \cdot \begin{cases} \displaystyle\sum_{\substack{k\neq l \\ i<k\leq v \\ v\leq l<j}} \sum_{d'_\ell \leq d_\ell} \sum_{\substack{d'_{\text{add}} \\ -b\leq d'_{\text{add}}\leq b}} \begin{pmatrix} Z^{I'}_{i,k}[d_\ell - d'_\ell] \cdot Z^{B,v}_{k,l}[d'_\ell, d'_\ell + d'_{\text{add}}] \\ \cdot Z^{I'}_{l,j}[(d_\ell + d_{\text{add}}) - (d'_\ell + d'_{\text{add}})] \end{pmatrix} & \text{if } -b < d_{\text{add}} < b \\ \displaystyle\sum_{\substack{k\neq l \\ i<k\leq v \\ v\leq l<j}} \sum_{d'_\ell \leq d_\ell} \sum_{\substack{d'_{\text{add}} \\ -b\leq d'_{\text{add}}\leq b}} \begin{pmatrix} Z^{I'}_{i,k}[d_\ell - d'_\ell] \cdot Z^{B,v}_{k,l}[d'_\ell, d'_\ell + d'_{\text{add}}] \\ \cdot Z^{I'_{\geq}}_{l,j}[(d_\ell + b) - (d'_\ell + d'_{\text{add}})] \end{pmatrix} & \text{if } d_{\text{add}} = b \\ \displaystyle\sum_{\substack{k\neq l \\ i<k\leq v \\ v\leq l<j}} \sum_{d'_\ell \leq d_\ell} \sum_{\substack{d'_{\text{add}} \\ -b\leq d'_{\text{add}}\leq b}} \begin{pmatrix} Z^{I'_{\geq}}_{i,k}[d_\ell - d'_\ell] \cdot Z^{B,v}_{k,l}[d'_\ell, d'_\ell + d'_{\text{add}}] \\ \cdot Z^{I'}_{l,j}[(d_\ell - b) - (d'_\ell + d'_{\text{add}})] \end{pmatrix} & \text{if } d_{\text{add}} = -b \end{cases}$$

References

1. D. H. Mathews, M. D. Disney, J. L. Childs, S. J. Schroeder, M. Zuker, and D. H. Turner. Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proc. Natl. Acad. Sci. USA*, 101:7287–7292, 2004.
2. M. Zuker and P. Stiegler. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Res.*, 9:133–148, 1981.
3. J. S. McCaskill. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29(6-7):1105–19, 1990.
4. R. Lorenz, S. H. Bernhart, C. Höner zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, and I. L. Hofacker. ViennaRNA Package 2.0. *Alg. Mol. Biol.*, 6:26, 2011.