An Algorithm for Computing Nucleic Acid Base-Pairing Probabilities Including Pseudoknots

ROBERT M. DIRKS,[1] NILES A. PIERCE[2]
[1]*Department of Chemistry, California Institute of Technology, Pasadena, California 91125*
[2]*Departments of Applied & Computational Mathematics and Bioengineering, California Institute of Technology, Mail Code 114-96, Pasadena, California 91125*

Received 21 January 2004; Accepted 19 March 2004
DOI 10.1002/jcc.20057
Published online in Wiley InterScience (www.interscience.wiley.com).

Abstract: Given a nucleic acid sequence, a recent algorithm allows the calculation of the partition function over secondary structure space including a class of physically relevant pseudoknots. Here, we present a method for computing base-pairing probabilities starting from the output of this partition function algorithm. The approach relies on the calculation of recursion probabilities that are computed by backtracking through the partition function algorithm, applying a particular transformation at each step. This transformation is applicable to any partition function algorithm that follows the same basic dynamic programming paradigm. Base-pairing probabilities are useful for analyzing the equilibrium ensemble properties of natural and engineered nucleic acids, as demonstrated for a human telomerase RNA and a synthetic DNA nanostructure.

© 2004 Wiley Periodicals, Inc. J Comput Chem 25: 1295–1304, 2004

Key words: DNA; RNA; base-pairing probabilities; partition function; pseudoknots

Introduction

Thermodynamic models based on nucleic acid secondary structure and nearest-neighbor identities[1–5] underly dynamic programming algorithms for predicting the minimum energy secondary structure[6–10] and calculating the partition function over secondary structure space.[10–12] In their original forms, these algorithms exclude the possibility of pseudoknots, a biologically relevant class of secondary structures[13] that also arises in DNA nanotechnology applications.[14,15] Pseudoknots result when two base pairs $i \cdot j$ and $d \cdot e$, with $i < d$, fail to satisfy the nesting property $i < d < e < j$ (see, e.g., Fig. 1). Recent extensions of the structure prediction[16–18] and partition function[18] algorithms allow the inclusion of certain pseudoknots.

For an ensemble of secondary structures $s \in \Omega$, the partition function

$$Q = \sum_{s \in \Omega} e^{-G_s/RT}$$

may be used to compute the probability

$$p(s^*) = \frac{1}{Q} e^{-G_{s^*}/RT} \tag{1}$$

that secondary structure s^* is sampled at thermodynamic equilibrium. The ensemble equilibrium can also be characterized by the matrix of base-pairing probabilities with entries $p_{i,j}$ corresponding to the probability that base i is paired with base j in Ω.

McCaskill's original article[11] defines elegant dynamic programs to compute the partition function and base-pairing probabilities over the ensemble of unpseudoknotted secondary structures. The partition function algorithm builds up recursively from short subsequences to the full strand, and then the pair probabilities are computed by working backwards to short subsequences using intermediate results from the partition function calculation. In the absence of pseudoknots, the partition function algorithm is sufficiently succinct that McCaskill is able to determine the form of the pair probability backtrack algorithm simply by considering the few possible forms of enclosing secondary structure for any given base pair. Although this approach is simple and efficient, it is not easily

Correspondence to: Niles A. Pierce; e-mail: niles@caltech.edu

Contract/grant sponsor: NSF graduate research fellowship (R.M.D.).

Contract/grant sponsor: Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory under F30602-010200561 (N.A.P.).

Contract/grant sponsor: Ralph M. Parsons Foundation (N.A.P.).

Contract/grant sponsor: Charles Lee Powell Foundation (N.A.P.).

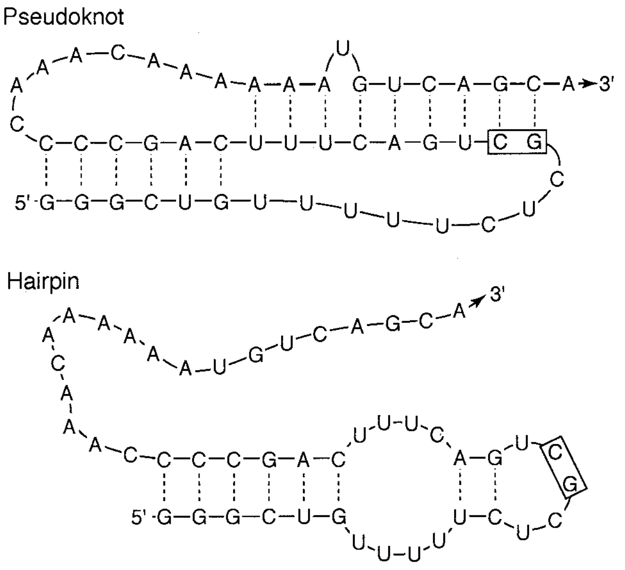


Figure 1. Secondary structures of competing pseudoknot and hairpin constructs in human telomerase RNA. The wild-type sequence is shown. For the two-point mutant implicated in dyskeratosis congenita, GC is replaced by AG in the shaded boxes, disrupting two base pairs in the pseudoknot construct. For the experimental studies of the hairpin structure,[20] the 18 nucleotides at the 3′ end are excluded to prevent formation of the pseudoknot.

generalizable to algorithmic extensions, such as the inclusion of pseudoknots. Here, we describe a general method for mechanically transforming the new pseudoknot partition function algorithm[18] to compute recursion probabilities, which can be used in turn to compute base-pairing probabilities. The transformation approach is generalizable to any future partition function extensions that follow the same dynamic programming paradigm.

Base-pairing probabilities assist in the analysis of biologically relevant pseudoknots. Here, we examine human telomerase RNA, which exists at equilibrium in both hairpin and pseudoknotted forms.[19] A two-point mutation, implicated in the disease dyskeratosis congenita, alters the thermodynamic balance between these competing structures.[20] This shift in equilibrium is clearly identifiable when the base-pairing probabilities for the two sequences are compared. Base-pairing probabilities that permit pseudoknots are also useful in analyzing synthetic DNA nanostructures.[14,15]

Algorithm

For clarity, we begin by considering the class of secondary structures excluding pseudoknots and then address the additional complexity that arises when pseudoknots are introduced.

Partition Function Recursions

For a strand of length N, the partition function may be computed over all unpseudoknotted secondary structures in $O(N^4)$ using the algorithm[10,11] summarized in Figure 2 (see ref. 18 for a detailed description in the same notation). [The complexity may be reduced to $O(N^3)$ by exploiting the formulation of the nearest-neighbor energy model for long interior loops.[18,21]] Partition function recursions are nonredundant in the sense that every secondary structure in the ensemble Ω is visited exactly once using a unique sequence of recursions. The algorithm computes the partition function $Q_{i,j}$ for each subsequence $[i, j]$ ignoring all bases exterior to $[i, j]$, starting from subsequences of length $l = 1$ and building up incrementally to $l = N$. The recursions that define $Q_{i,j}$ rely on additional restricted partition functions $Q^b_{i,j}$ and $Q^m_{i,j}$. $Q^b_{i,j}$ represents the partition function for subsequence $[i, j]$ given that i and j are base paired and $Q^m_{i,j}$ is used to calculate multiloop contributions. At the end of the recursive process, the full partition function Q is given by $Q_{1,N}$ and the values of $Q_{i,j}$, $Q^b_{i,j}$, $Q^m_{i,j}$ are stored in matrices for $1 \le i, j \le N$. These intermediate results will play a critical role in the new algorithm described below.

Recursion Probabilities

Following the execution of the partition function calculation, a second algorithm can be implemented to calculate probability matrices, P, P^b, P^m, corresponding to the Q, Q^b, Q^m matrices. The values stored in these P-type matrices will be termed *recursion probabilities.*

Recursion probabilities can be intuitively described as follows. Consider sampling the ensemble of secondary structures $s \in \Omega$ where the probability of selecting structure s^* is given by the Boltzmann probability (1). For each secondary structure s^*, the contribution to Q is computed by a unique recursion sequence involving specific $Q_{i,j}$, $Q^b_{i,j}$, and $Q^m_{i,j}$ intermediates. Associating these intermediates with structure s^*, the recursion probability $P_{i,j}$, $P^b_{i,j}$ or $P^m_{i,j}$ corresponds to the probability that the sampled structure s^* requires the use of the corresponding intermediate $Q_{i,j}$, $Q^b_{i,j}$ or $Q^m_{i,j}$ to calculate the partition function contribution.

Recent work by Ding and Lawrence[22] exploits quantities related to recursion probabilities to statistically sample the distribution of unpseudoknotted secondary structures for a given sequence. Here, we develop a general approach for computing P-type ma-

```
Initialize $(Q, Q^b, Q^m)$ // $O(N^2)$ space
Set all values to 0 except $Q_{i,i-1} = 1$
for $l = 1, N$
  for $i = 1, N-l+1$
    $j = i+l-1$
    // $Q^b$ recursion
    $Q^b_{i,j} = \exp\{-G^{\text{hairpin}}_{i,j}/RT\}$
    for $d = i+1, j-5$ // loop over all possible rightmost pairs $d \cdot e$
      for $e = d+4, j-1$
        $Q^b_{i,j}$ += $\exp\{-G^{\text{interior}}_{i,d,e,j}/RT\}\, Q^b_{d,e}$
        $Q^b_{i,j}$ += $Q^m_{i+1,d-1}\, Q^b_{d,e}\, \exp\{-[\alpha_1 + 2\alpha_2 + \alpha_3(j-e-1)]/RT\}$
    // $Q, Q^m$ recursions
    $Q_{i,j} = 1$ //empty recursion
    for $d = i, j-4$ // loop over all possible rightmost pairs $d \cdot e$
      for $e = d+4, j$
        $Q_{i,j}$+= $Q_{i,d-1}\, Q^b_{d,e}$
        $Q^m_{i,j}$ += $\exp\{-[\alpha_2 + \alpha_3(d-i) + \alpha_3(j-e)]/RT\}\, Q^b_{d,e}$
        $Q^m_{i,j}$ += $Q^m_{i,d-1}\, Q^b_{d,e}\, \exp\{-[\alpha_2 + \alpha_3(j-e)]/RT\}$
//Partition function is $Q_{1,N}$
```

Figure 2. $O(N^4)$ partition function algorithm that excludes pseudoknots.

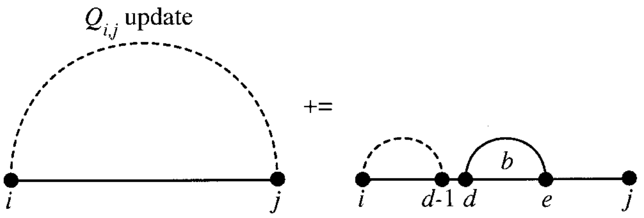


Figure 3. Recursion diagram corresponding to recursive update (2), depicting the addition to $Q_{i,j}$ of partition function contributions for those structures with rightmost base pair $d \cdot e$. See ref. 18 for a thorough description of the partition function algorithm (with or without pseudoknots) in terms of recursion diagrams.

trices given a set of Q-type matrices and corresponding partition function recursions.

An algorithm for computing recursion probabilities can be formulated in a mechanical way starting from a set of partition function recursions. The crux of this formulation is the repeated application of a single transformation to the partition function code. In particular, updates of the form

$$Q_{i,j} \mathrel{+}= Q_{i,d-1}Q^b_{d,e} \tag{2}$$

(equivalent to the recursion diagram of Fig. 3) are converted to the following series of statements

$$\Delta p = P_{i,j}\ \overbrace{Q_{i,d-1}Q^b_{d,e}/Q_{i,j}}^{\text{conditional probability}}$$

$$P_{i,d-1} \mathrel{+}= \Delta p$$

$$P^b_{d,e} \mathrel{+}= \Delta p \tag{3}$$

Specifically, the right-hand side (RHS) of each recursive update is divided by the left-hand side (LHS), and the P term corresponding to the new denominator is multiplied by this quotient. The resulting probabilities, temporarily stored as Δp, are subsequently added to every P-type value corresponding to the Q-type terms on the RHS of the original statement (2).

To understand this transformation, recall that $Q_{i,j}$, $Q^b_{i,j}$ and $Q^m_{i,j}$ are partition functions for structural subclasses of the full sequence. In recursive updates such as (2), the ratio of the RHS to the fully computed LHS corresponds to the probability that a structure drawn from an equilibrium ensemble defined by the LHS partition function is in the subensemble defined by the RHS partition function. As an example, transformation (3) states that for any i, d, e, j, the structures represented by $Q_{i,j}$ partially consist of substructures represented by $Q_{i,d-1}$ and $Q^b_{d,e}$. Consequently, once the probability $P_{i,j}$ is determined, it can be used to augment $P_{i,d-1}$ and $P^b_{d,e}$ because the frequencies of the corresponding substructures within the $Q_{i,j}$ ensemble can be derived from $Q_{i,d-1}$ and $Q^b_{d,e}$. By backtracking through the partition function algorithm and transforming all recursive updates analagously to (3), probabilities can be calculated for each recursion.

Starting from the partition function algorithm of Figure 2, the recursion probability algorithm is obtained by performing three modifications: (1) the two outermost loops are altered so that the algorithm starts with the full strand of length $l = N$ and decrements down to subsequences of length $l = 1$; (2) all recursive updates are transformed as for (3) above; (3) the order of the recursion blocks (Q^b, [Q, Q^m]) is reversed ([P, P^m], P^b). This last modification is necessary because the recursion order in the partition function algorithm ensures that if one quantity (e.g., $Q_{i,j}$) recurses to another quantity of the same length (e.g., $Q^b_{i,j}$) then the "lower level" quantity (i.e., $Q^b_{i,j}$) is calculated first. The reverse ordering is needed for the recursion probability algorithm, because $P^b_{i,j}$ cannot be used until it has been fully computed in the $P_{i,j}$ loop.

The pseudocode in Figure 4 details the outcome of these transformations for the unpseudoknotted case. This modified algorithm reverses the flow of the partition function calculation and incrementally determines all recursion probabilities (frequencies of families of structures), based on the probabilities of all superstructures that directly contain them. Once recursion probabilities are computed for all i and j, the base-pairing probability $p_{i,j}$ is simply $P^b_{i,j}$, because $Q^b_{i,j}$ is associated with every structure s in which $i \cdot j$ appears, and $i \cdot j$ is associated with exactly one $Q^b_{i,j}$. By starting from a more complicated $O(N^3)$ partition function algorithm,[18,21] the computational complexity of the recursion probability algorithm can also be reduced to $O(N^3)$ as described in the Appendix.

Pseudoknots

The procedure outlined above for obtaining recursion probability algorithms is equally applicable to a new partition function algorithm that includes pseudoknots (see the pseudocode of Fig. 21 in ref. 18). For the unpseudoknotted algorithm, all base pairs stem

```
Compute $Q, Q^b, Q^m$ using $O(N^4)$ partition function algorithm
Initialize $(P, P^b, P^m)$ // $O(N^2)$ space
Set all $P$-type values to 0
$P_{1,N} = 1$ //probability of "recursing" to the entire strand is 1
for $l = N, 1$ //decrement
  for $i = 1, N-l+1$
    $j = i+l-1$
    //$P, P^m$ recursions
    for $d = i, j-4$ // loop over all possible rightmost pairs $d \cdot e$
      for $e = d+4, j$
        $\Delta p = P_{i,j} Q_{i,d-1}\, Q^b_{d,e}/Q_{i,j}$
        $P_{i,d-1} \mathrel{+}= \Delta p$
        $P^b_{d,e} \mathrel{+}= \Delta p$
        $P^b_{d,e} \mathrel{+}= P^m_{i,j} \exp\{-[\alpha_2 + \alpha_3(d-i) + \alpha_3(j-e)]/RT\}\, Q^b_{d,e}/Q^m_{i,j}$
        $\Delta p = P^m_{i,j} Q^m_{i,d-1}\, Q^b_{d,e} \exp\{-[\alpha_2 + \alpha_3(j-e)]/RT\}/Q^m_{i,j}$
        $P^m_{i,d-1} \mathrel{+}= \Delta p$
        $P^b_{d,e} \mathrel{+}= \Delta p$
    // $P^b$ recursion
    for $d = i+1, j-5$ // loop over all possible rightmost pairs $d \cdot e$
      for $e = d+4, j-1$
        $P^b_{d,e} \mathrel{+}= P^b_{i,j}\, Q^b_{d,e} \exp\{-G^{\text{interior}}_{i,d,e,j}/RT\}/Q^b_{i,j}$
        $\Delta p = P^b_{i,j} Q^m_{i+1,d-1}\, Q^b_{d,e} \exp\{-[\alpha_1 + 2\alpha_2 + \alpha_3(j-e-1)]/RT\}/Q^b_{i,j}$
        $P^m_{i+1,d-1} \mathrel{+}= \Delta p$
        $P^b_{d,e} \mathrel{+}= \Delta p$
```

Figure 4. $O(N^4)$ recursion probability algorithm that excludes pseudoknots. For simplicity, we omit details such as checking for updates with zero in the denominator (in which case the numerator will also evaluate to zero and the expression should be skipped).

from Q^b recursions, so the values stored in P^b are precisely the desired probabilities (i.e., $p_{i,j} = P^b_{i,j}$). For the pseudoknotted case, $P^b_{i,j}$ only gives the probability that i and j form a nested pair. The full base-pairing probability must also take into consideration those base pairs that are nonnested and lead to pseudoknotted structures (termed gap-spanning pairs in ref. 18). For these gap-spanning pairs, there is no single recursion probability that represents the contribution to $p_{i,j}$. However, this contribution may be succinctly represented in terms of Q-type and P-type matrices for the full pseudoknotted algorithm.

A new set of quantities, $P^{bg}_{i,j}$, will be used to store the base pairing probabilities of $i \cdot j$ gap-spanning pairs in pseudoknots. The most pertinent recursion probability, $P^g_{i,d,e,j}$, stores the probability of a gap structure with outer gap-spanning pair $i \cdot j$ and inner gap-spanning pair $d \cdot e$ corresponding to the partition function recursion $Q^g_{i,d,e,j}$ (see Fig. 19 in ref. 18). Due to the structure of the $Q^g_{i,d,e,j}$ recursion, the sum of $P^g_{i,d,e,j}$ over all values of d, e precisely gives the probability of an outer pair $i \cdot j$

$$P^{bg}_{i,j} \mathrel{+}= \sum_{i<d<e<j} P^g_{i,d,e,j}. \tag{4}$$

However, the sum of $P^g_{i,d,e,j}$ over all values of i, j does *not* give the probability of an inner pair $d \cdot e$, because the same inner pair may be present in multiple recursions required to define the same secondary structure. To correctly determine the probabilities of inner gap-spanning pairs, only the portion of P^g that corresponds to calling Q^g directly from Q^{gl} should be included

$$P^{bg}_{d,f} \mathrel{+}= \sum_{i<d\leq e<f<j} P^{gl}_{i,e,f,j} Q^g_{i,d,f,j} Q^z_{d+1,e} \exp(-\beta_2/RT)/Q^{gl}_{i,e,f,j}. \tag{5}$$

Here, Q^{gl} and Q^z are partition function recursions used to define the interior structure of a pseudoknot, and β_2 is a pseudoknot energy parameter (see Figs. 18 and 12 in ref. 18). Allowing pseudoknots, the total probability of a base pair $i \cdot j$ is then

$$p_{i,j} = P^b_{i,j} + P^{bg}_{i,j}.$$

Pseudocode detailing the algorithm for computing recursion probabilities in the pseudoknotted case is provided in Figure 5, where the calculation of $P^{bg}_{i,j}$ using (4) and (5) has been embedded at little additional cost. [Note that (4) and (5) use different indices for P^{bg} to maintain consistency with the pseudocode.] In the Appendix, we describe how to reduce the complexity of the pseudoknotted algorithm from $O(N^6)$ to $O(N^5)$.

Methods

The standard energy model[4] and pseudoknot extension[18] are implemented as described previously,[18] including dangle energies and penalties for helices not terminated by a G • C pair. These terms do not change the structure of the recursions described in the pseudocode and are omitted for clarity. Coaxial stacking contributions are not included in the physical model, as it is unclear how to treat different stackings associated with the same secondary structure in the context of the partition function. To maintain consistency with a recent design study,[23] dangle energies are treated analogously to the d2 option in the Vienna package.[10] Following this approach, dangle energies are included even if two helices are separated by one or zero bases, providing some compensation for the neglect of coaxial stacking bonuses.

Applications

The recursion probability algorithm provides a simple, general method for calculating the frequency of various substructures in the ensemble of states for a given nucleic acid. Base-pairing probabilities derived from the recursion probabilities are particularly useful for analyzing secondary structure via dot plot analyses.[11] A traditional dot plot depicts the probabilities of forming all possible $i \cdot j$ base pairs. In the case of pseudoknots, the dot plot can be decomposed into two dot plots—one for nested pairs and one for nonnested gap-spanning pairs.

To see the utility of this decomposition, calculations were run on wild-type and mutant sequences of a pseudoknot construct derived from human telomerase RNA.[20] Experimental evidence suggests that this pseudoknot exists in equilibrium with an alternative, hairpin structure, and that this equilibrium functions as a biological switch.[19] A two-point mutant, found in a small percentage of people, shifts the equilibrium towards the hairpin structure, leading to a disease known as dyskeratosis congenita.[19] Feigon and coworkers[20] examine this shift in equilibrium for segments of the wild-type and mutant sequences described in Figure 1, revealing that the pseudoknot conformation dominates the hairpin for the wild-type sequence (~95% to ~5%) but competes roughly equally in the mutant sequence (~50% to ~50%). Using preliminary pseudoknot parameters,[18] energies were computed for both the wild-type sequence and the two-point mutant on the pseudoknotted and hairpin structures. The predicted energies in Table 1 match reasonably well with experimental values.[20] For the wild-type sequence, the disparity between the pseudoknot and hairpin energies suggests an equilibrium that favors the more stable pseudoknot. In contrast, the energies for the double mutant sequence suggest a more balanced equilibrium. Figures 6 and 7 illustrate that the hairpin conformation has a significant impact on the pair probabilities for the mutant, but not for the wild-type sequence.

Base-pairing probabilities can also be used to construct metrics for evaluating nucleic acid designs. The secondary structure s may be described by a symmetric $N \times N$ matrix S with entries $S_{i,j} = 1$ if s contains base pair $i \cdot j$ and $S_{i,j} = 0$ otherwise. We augment this matrix by an additional column with entries $S_{i,N+1} = 1$ if base i is unpaired and $S_{i,N+1} = 0$ otherwise. Hence, each row sum is one. For a given sequence of length N, the metric[23]

$$n(s^*) = N - \sum_{\substack{1\leq i\leq N \\ 1\leq j\leq N+1}} p_{i,j} S^*_{i,j}$$

represents the average number of nucleotides that differ from the target secondary structure s^* at thermodynamic equilibrium. This

```
Compute $Q, Q^b, Q^m, Q^p, Q^z, Q^g, Q^{gl}, Q^{gr}, Q^{gls}, Q^{grs}$
// using $O(N^5)$ partition function algorithm
Initialize $(P, P^b, P^m, P^p, P^z, P^{bg})$ // $O(N^2)$ space
Initialize $(P^g, P^{gl}, P^{gr}, P^{gls}, P^{grs})$ // $O(N^4)$ space
// Initialize $(Q^x, Q^{x1}, Q^{x2}, P^x, P^{x1}, P^{x2})$ for $O(N^5)$ version
// Set all $Q^x$-type values to 0 for $O(N^5)$ version
Set all $P$-type values to 0
$P_{1,N} = 1$ //probability of "recursing" to the entire strand is 1
for $l = N, 1$ //decrement
  // Initialize $Q^x = Q^{x1}$, $Q^{x1} = Q^{x2}$, $Q^{x2} = 0$ for $O(N^5)$ version
  // Initialize $P^x = P^{x1}$, $P^{x1} = P^{x2}$, $P^{x2} = 0$ for $O(N^5)$ version
  for $i = 1, N-l+1$
    $j = i+l-1$
    //$P, P^m, P^z$ recursions
    for $d = i, j-4$ // all possible rightmost pairs $d{\cdot}e$
      for $e = d+4, j$
        $\Delta p = P_{i,j} Q_{i,d-1}\, Q^b_{d,e} / Q_{i,j}$
        $P_{i,d-1} \mathrel{+}= \Delta p$
        $P^b_{d,e} \mathrel{+}= \Delta p$
        $P^b_{d,e} \mathrel{+}= P^m_{i,j} \exp\{-[\alpha_2 + \alpha_3(d-i) + \alpha_3(j-e)]/RT\}\, Q^b_{d,e}/Q^m_{i,j}$
        $\Delta p = P^m_{i,j} Q^m_{i,d-1}\, Q^b_{d,e} \exp\{-[\alpha_2 + \alpha_3(j-e)]/RT\}/Q^m_{i,j}$
        $P^m_{i,d-1} \mathrel{+}= \Delta p$
        $P^b_{d,e} \mathrel{+}= \Delta p$
        $\Delta p = P^z_{i,j} Q^z_{i,d-1} Q^b_{d,e} \exp\{-[\beta_2 + \beta_3(j-e)]/RT\}/Q^z_{i,j}$
        $P^z_{i,d-1} \mathrel{+}= \Delta p$
        $P^b_{d,e} \mathrel{+}= \Delta p$
    for $d = i, j-8$ // all possible rightmost pseudoknots filling $[d, e]$
      for $e = d+8, j$
        $\Delta p = P_{i,j} Q_{i,d-1} Q^p_{d,e} \exp\{-\beta_1/RT\}/Q_{i,j}$
        $P_{i,d-1} \mathrel{+}= \Delta p$
        $P^p_{d,e} \mathrel{+}= \Delta p$
        $P^p_{d,e} \mathrel{+}= P^m_{i,j} \exp\{-[\beta^m_1 + 2\alpha_2 + \alpha_3(d-i+j-e)]/RT\} Q^p_{d,e}/Q^m_{i,j}$
        $\Delta p = P^m_{i,j} Q^m_{i,d-1} Q^p_{d,e} \exp\{-[\beta^m_1 + 2\alpha_2 + \alpha_3(j-e)]/RT\}/Q^m_{i,j}$
        $P^m_{i,d-1} \mathrel{+}= \Delta p$
        $P^p_{d,e} \mathrel{+}= \Delta p$
        $\Delta p = P^z_{i,j} Q^z_{i,d-1} Q^p_{d,e} \exp\{-[\beta^p_1 + 2\beta_2 + \beta_3(j-e)]/RT\}/Q^z_{i,j}$
        $P^z_{i,d-1} \mathrel{+}= \Delta p$
        $P^p_{d,e} \mathrel{+}= \Delta p$
    //$P^p$ recursion
    for $d = i+2, j-4$
      for $e = \max(d+2, i+5), j-3$
        for $f = e+1, j-2$
          $\Delta p = P^p_{i,j} Q^{gl}_{i,d-1,e,f} Q^{gr}_{d,e-1,f+1,j}/Q^p_{i,j}$
          $P^{gl}_{i,d-1,e,f} \mathrel{+}= \Delta p$
          $P^{gr}_{d,e-1,f+1,j} \mathrel{+}= \Delta p$
    // $P^{gr}$ recursion
    for $d = i+1, j-4$
      for $e = d+3, j-1$
        for $f = e, j-1$
          $\Delta p = P^{gr}_{i,d,e,j} Q^{gl}_{i,d,f,j} Q^z_{e,f-1}/Q^{gr}_{i,d,e,j}$
          $P^{gl}_{i,d,f,j} \mathrel{+}= \Delta p$
          $P^z_{e,f-1} \mathrel{+}= \Delta p$
    // $P^{gl}$ recursion
    for $d = i+1, j-5$
      for $f = d+4, j-1$
        for $e = d, f-3$
          $\Delta p = P^{gl}_{i,e,f,j} Q^g_{i,d,f,j} Q^z_{d+1,e} \exp(-\beta_2/RT)/Q^{gl}_{i,e,f,j}$
          $P^g_{i,d,f,j} \mathrel{+}= \Delta p$
          $P^z_{d+1,e} \mathrel{+}= \Delta p$
          $P^{bg}_{d,f} \mathrel{+}= \Delta p$ // $P^{bg}$ inner gap-spanning base-pairing prob
    // $P^{grs}$ recursion
    for $d = i+1, j-10$
      for $e = d+4, j-6$
        for $f = e+1, j-5$
          $\Delta p = P^{grs}_{i,d,e,j} Q^g_{i,d,e,f} Q^m_{f+1,j} \exp(-\alpha_2/RT)/Q^{grs}_{i,d,e,j}$
          $P^g_{i,d,e,f} \mathrel{+}= \Delta p$
          $P^m_{f+1,j} \mathrel{+}= \Delta p$
    // $P^{gls}$ recursion
    for $c = i+5, j-6$
      for $d = c+1, j-5$
        for $e = d+4, j-1$
          $\Delta p = P^{gls}_{i,d,e,j} \exp(-\alpha_2/RT) Q^m_{i,c-1} Q^g_{c,d,e,j}/Q^{gls}_{i,d,e,j}$
          $P^m_{i,c-1} \mathrel{+}= \Delta p$
          $P^g_{c,d,e,j} \mathrel{+}= \Delta p$
    // $P^g$ recursion
    for $d = i+2, j-6$ // simple interior loops
      for $e = d+4, j-2$
        for $c = i+1, d-1$
          for $f = e+1, j-1$
            $P^g_{c,d,e,f} \mathrel{+}= P^g_{i,d,e,j} \exp(-G^{\text{interior}}_{i,c,f,j}/RT) Q^g_{c,d,e,f}/Q^g_{i,d,e,j}$
    // for $O(N^5)$ version, replace previous five lines with:
    // call fastiloopsN5$(i, j, l, Q^g, Q^x, Q^{x2}, P^g, P^x, P^{x2})$
    for $d = i+6, j-5$
      for $e = d+4, j-1$
        $\Delta p = P^g_{i,d,e,j} Q^m_{i+1,d-1} \exp\{-[\alpha_1 + 2\alpha_2 + \alpha_3(j-e-1)]/RT\}/Q^g_{i,d,e,j}$
        $P^m_{i+1,d-1} \mathrel{+}= \Delta p$
    for $d = i+1, j-10$
      for $e = d+4, j-6$
        $\Delta p = P^g_{i,d,e,j} \exp\{-[\alpha_1 + 2\alpha_2 + \alpha_3(d-i-1)]/RT\} Q^m_{e+1,j-1}/Q^g_{i,d,e,j}$
        $P^m_{e+1,j-1} \mathrel{+}= \Delta p$
    for $d = i+6, j-10$
      for $e = d+4, j-6$
        $\Delta p = P^g_{i,d,e,j} Q^m_{i+1,d-1} \exp\{-[\alpha_1 + 2\alpha_2]/RT\} Q^m_{e+1,j-1}/Q^g_{i,d,e,j}$
        $P^m_{i+1,d-1} \mathrel{+}= \Delta p$
        $P^m_{e+1,j-1} \mathrel{+}= \Delta p$
    for $d = i+7, j-6$
      for $e = d+4, j-2$
        for $f = e+1, j-1$
          $\Delta p = P^g_{i,d,e,j} Q^{gls}_{i+1,d,e,f} \exp\{-[\alpha_1 + \alpha_2 + \alpha_3(j-f-1)]/RT\}/Q^g_{i,d,e,j}$
          $P^{gls}_{i+1,d,e,f} \mathrel{+}= \Delta p$
    for $d = i+2, j-11$
      for $e = d+4, j-7$
        for $c = i+1, d-1$
          $\Delta p = P^g_{i,d,e,j} \exp\{-[\alpha_1 + \alpha_2 + \alpha_3(c-i-1)]/RT\} Q^{grs}_{c,d,e,j-1}/Q^g_{i,d,e,j}$
          $P^{grs}_{c,d,e,j-1} \mathrel{+}= \Delta p$
    for $d = i+7, j-11$
      for $e = d+4, j-7$
        for $c = i+6, d-1$
          $\Delta p = P^g_{i,d,e,j} Q^m_{i+1,c-1} Q^{grs}_{c,d,e,j-1} \exp\{-[\alpha_1 + \alpha_2]/RT\}/Q^g_{i,d,e,j}$
          $P^m_{i+1,c-1} \mathrel{+}= \Delta p$
          $P^{grs}_{c,d,e,j-1} \mathrel{+}= \Delta p$
    // $P^{bg}$ outer gap-spanning base-pairing prob
    for $d = i+1, j-5$
      for $e = d+4, j-1$
        $P^{bg}_{i,j} + = P^g_{i,d,e,j}$
    // $P^b$ recursion
    for $d = i+1, j-5$ // all possible rightmost pairs $d{\cdot}e$
      for $e = d+4, j-1$
        $P^b_{d,e} \mathrel{+}= P^b_{i,j} \exp(-G^{\text{interior}}_{i,d,e,j}/RT) Q^b_{d,e}/Q^b_{i,j}$
        $\Delta p = P^b_{i,j} Q^m_{i+1,d-1}\, Q^b_{d,e} \exp\{-[\alpha_1 + 2\alpha_2 + \alpha_3(j-e-1)]/RT\}/Q^b_{i,j}$
        $P^m_{i+1,d-1} \mathrel{+}= \Delta p$
        $P^b_{d,e} \mathrel{+}= \Delta p$
    for $d = i+1, j-9$ // all possible rightmost pseudoknots filling $[d, e]$
      for $e = d+8, j-1$
        $G^{\text{recursion}} = \alpha_1 + \beta^m_1 + 3\alpha_2 + \alpha_3(j-e-1)$
        $P^p_{d,e} \mathrel{+}= P^b_{i,j} \exp\{-[G^{\text{recursion}} + \alpha_3(d-i-1)]/RT\} Q^p_{d,e}/Q^b_{i,j}$
        $\Delta p = P^b_{i,j} Q^m_{i+1,d-1}\, Q^p_{d,e} \exp\{-G^{\text{recursion}}/RT\}/Q^b_{i,j}$
        $P^m_{i+1,d-1} \mathrel{+}= \Delta p$
        $P^p_{d,e} \mathrel{+}= \Delta p$
```

Figure 5. $O(N^6)$ recursion probability algorithm that includes a class of pseudoknots. Modifications required to produce an $O(N^5)$ version of the algorithm are noted in comments. See the Appendix for details.

Table 1. Energy Comparisons for Human Telomerase RNA Constructs.

		Energies (kcal/mol)	
RNA	Conformation	ΔG_{exp}	ΔG_{calc}
Wild-type	Pseudoknot	−17.8	−18.5[b]
	Hairpin	−9.8[a]	−11.5[c]
Mutant	Pseudoknot	−11.2	−11.3[b]
	Hairpin	−10.5[a]	−11.5[c]

[a] Experiments were performed on partial sequences that excluded the 18 nucleotides on the 3′ end to prevent the formation of pseudoknots.[20] This truncation does not affect the corresponding ΔG_{calc}.

[b] A related pseudoknot structure that is otherwise identical but omits the three consecutive $A \cdot U$ pairs in the stem with the bulge loop is predicted to be 0.5 kcal/mol more stable.

[c] The secondary structure energy calculation ignores the four consecutive noncanonical base pairs that are observed to close the interior loop in the hairpin stem.[20]

is a less stringent metric than $p(s^*)$, the probability that the sequence exactly adopts structure s^*; even if $p(s^*)$ is not close to unity, $n(s^*)$ can still be small if the equilibrium ensemble is dominated by structures that differ only slightly from s^*.

It is illustrative to compare the two metrics on a real design problem involving pseudoknots. For example, Winfree et al.[14] designed and constructed DNA double-crossover molecules[24] that interact to form a two-dimensional lattice with a pseudoknotted unit cell. These sequence designs were performed using sequence symmetry minimization[25] to ensure that incorrectly paired subsequences of length six would always contain at least one mismatch and most incorrectly paired subsequences of length five would also contain a mismatch.[14] Lacking DNA pseudoknot parameters, we examine an RNA analog of their sequence for the portion of the pseudoknotted unit cell depicted in Figure 8a. The probability of adopting the target structure is $p(s^*) = 0.1$ and the average number of incorrect nucleotides is $n(s^*) = 4.0$. The low value of $p(s^*)$ might possibly indicate a cause for concern, but for a structure with 90 nucleotides and helices of length eight, the average number of incorrect nucleotides is relatively small. Hence, it is not surprising that the sequence behaves well experimentally, demonstrating the correct base-pairing topology despite slight predicted variations at the ends of helices. The dot plot in Figure 8b illustrates the similarity between the average structure and the desired target.

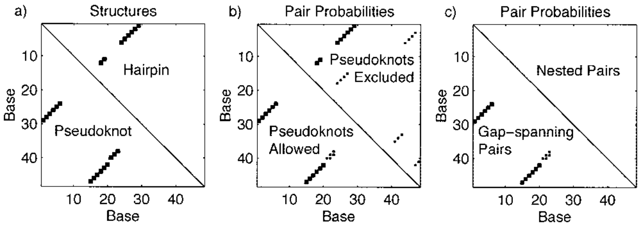


Figure 6. Dot plots for wild-type human telomerase RNA. (a) Pseudoknot (bottom left) and hairpin (top right) constructs. For (b) and (c), large dots indicate a $p_{i,j} \geq 0.5$ and small dots indicate $0.5 > p_{i,j} \geq 0.05$. (b) Base-pairing probabilities including pseudoknots (bottom left) and excluding pseudoknots (top right). (c) A decomposition of the full base-pairing probabilities into gap-spanning pairs (bottom left) and nested pairs (top right). Note that there are no nested pairs with significant probability, indicating that pseudoknot conformations are dominating the equilibrium.

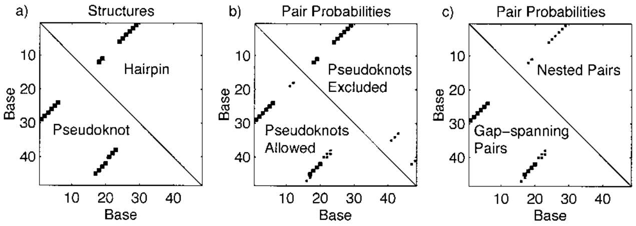


Figure 7. Dot plots for double mutant human telomerase RNA. The plots are analogous to those of Figure 6. The key difference is observed in (c), where the hairpin stem appears as both gap-spanning pairs and nested pairs, indicating the increased significance of hairpin conformations.

Interestingly, design methods described in previous work[23] can be used, in conjunction with the pseudoknot partition function algorithm, to find sequences that achieve $p(s^*) = 0.98$ and $n(s^*) \ll 1$. It is unclear whether these sequences would provide any experimental benefit for this system (even given a perfect energy model), because the difference between $n(s^*) \approx 4$ and $n(s^*) \ll 1$ may be lost in experimental noise. By contrast, if a sequence produced $p(s^*) = 0.1$ with $n(s^*) \gg 4$, then the equilibrium ensemble could include important structures differing significantly from the target structure.

Conclusions

A general transformation rule extends nucleic acid partition function algorithms to calculate recursion probabilities, which in turn, can be used to compute base-pairing probabilities. We use this approach to derive an algorithm for computing base-pairing probabilities starting from a partition function algorithm that includes a class of pseudoknots. The same strategy will apply to future partition function extensions that follow the same dynamic programming paradigm.

To demonstrate the utility of base-pairing probabilities, calculations were performed on a pseudoknot/hairpin construct thought to represent an important biological switch. In agreement with experimental evidence, the computational results indicate that the pseudoknot dominates the hairpin for the wild-type sequence, but not for the double mutant. Base-pairing probabilities were also used to examine the ensemble properties of a synthetic nucleic acid sequence designed to assemble into a pseudoknotted double-crossover molecule. The average number of incorrect nucleotides was found to be small, suggesting that the relatively low computed probability of adopting the

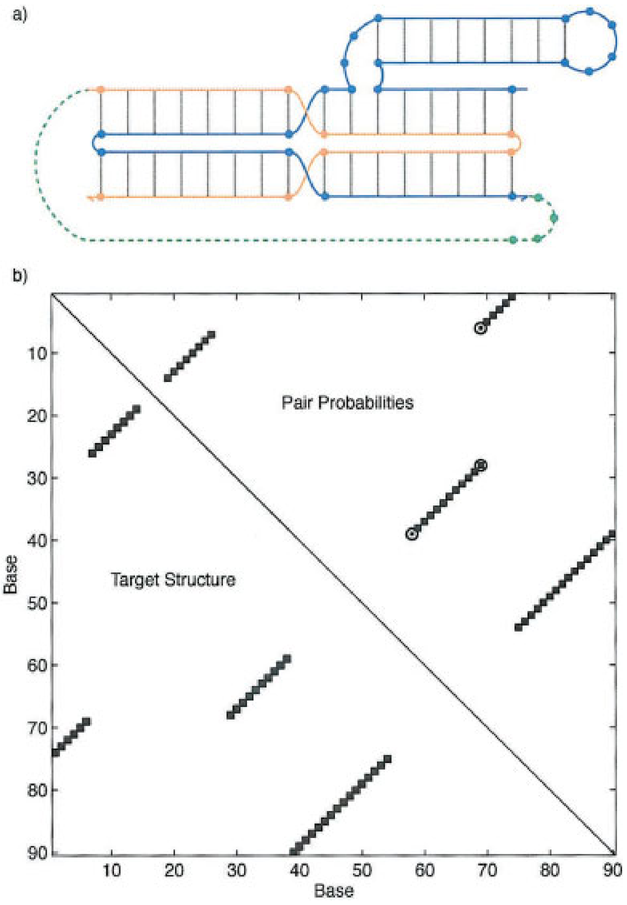


Figure 8. Computational examination of a pseudoknotted DNA nanostructure. (a) Secondary structure for a double-crossover molecule that forms a portion of the unit cell in a two-dimensional lattice.[14] For our computational study, we join the blue and orange strands (arrows denote 3′) into a single strand using auxiliary nucleotides (green) to facilitate the use of the single-stranded partition function algorithm.[18] In the absence of DNA pseudoknot parameters, we consider the RNA analog 5′-CCAACUCCUAGCGAUUUUUCGCUAGGUUUACCA-GAUCCACAAGCCGACGUUACA-UUUU-GGAUCUGGUAAG-UUGGUGUAACGUCGGCUUGU-3′, where the interior hyphens denote the boundaries of the auxiliary linker segment. (b) Dot plot analysis of the designed sequence. The bottom left depicts the base-pairs in the target structure, and the upper right depicts the base-pairing probabilities. Large dots indicate a $p_{i,j} \geq 0.5$ and small dots indicate $0.5 > p_{i,j} \geq 0.05$. The circles indicate the major differences between the target structure and the calculated pair probabilities.

target secondary structure should not significantly affect the experimental performance of the molecule.

Software Download

The algorithms described in this article are available for download at http://www.acm.caltech.edu/~niles as part of the NUPACK software suite.

Acknowledgments

We wish to thank C. Ueda for discussions on human telomerase RNA and E. Winfree for discussions on the DNA lattice.

Appendix: Reducing Computational Complexity

The algorithms presented in the main text provide an inefficient treatment of interior loops. By exploiting the form of the interior loop potential function, the computational complexity of the partition function algorithms excluding and including pseudoknots can be reduced by a factor of N, where N is the sequence length.[18,21] A detailed description of the "fastiloops" treatment is provided in ref. 18 and the corresponding Supplementary Material. The "fastiloops" modification detracts from the simplicity of the presentation because the necessary recursions do not conform to the same structure as the other terms in the algorithm. Here, we describe the extension of this approach to recursion probability algorithms.

In the unpseudoknotted case, pseudocode for an $O(N^3)$ partition function algorithm is provided in Figure 11 of ref. 18, which employs the "fastiloops" function of Supplementary Material Figure S2. To this point, we have assumed that all Q-type values are accessible at the end of the partition function calculation. For the "fastiloops" methods, the values Q^x, Q^{x1} and Q^{x2} are computed on the fly and discarded to save memory. Hence, for the recursion probability algorithm, it is necessary to recompute the Q^x-type terms at the same time that the corresponding P^x-type terms are calculated. An $O(N^3)$ recursion probability algorithm that excludes pseudoknots is described in Figure A1, which references the function "fastiloopsN3" of Figure A2. If pseudoknots are included, the computational complexity of the recursion probability algorithm in Figure 5 is reduced to $O(N^5)$ using "fastiloopsN5" described in Figure A3. A few aspects of the "fastiloopsN3" and "fastiloopsN5" routines deserve mention. It is advisable to review the relevant sections of ref. 18 and the corresponding Supplementary Material before proceeding.

An interior loop with closing pair $i \cdot j$ and interior pair $d \cdot e$ has energy $G^{\text{interior}}_{i,d,e,j}$, sides of lengths

$$L_1 \equiv (d - i - 1), \quad L_2 \equiv (j - e - 1), \tag{6}$$

and size $L_1 + L_2$. Loops with both $L_1 \geq 4$ and $L_2 \geq 4$ are termed "extensible" and their contributions to the partition function algorithm are calculated using Q^x. Furthermore, Q^x also contains information about "possible extensible loops" for which the definitions of L_1, L_2 are the same but i and j are not required to base-pair.

The partition function algorithm examines subsequences of length $l = j - i + 1$, starting with $l = 1$ and ending with $l = N$. Q^x is efficiently calculated using the extension identity [see eq. (15) of ref. 18],

$$Q^x_{i-1,s+2} = \Gamma(s+2)\Big|^{L_1=4}_{L_1+L_2=s+2} + \Gamma(s+2)\Big|^{L_2=4}_{L_1+L_2=s+2} + [Q^x_{i,s}\exp\{-[\gamma_1(s+2) - \gamma_1(s)]/RT\}] \tag{7}$$

```
Compute $Q, Q^b, Q^m, Q^s, Q^{ms}$ using $O(N^3)$ partition function algorithm
Initialize $(Q^x, Q^{x1}, Q^{x2}, P, P^b, P^m, P^s, P^{ms}, P^x, P^{x1}, P^{x2})$ // $O(N^2)$ space
Set all $Q^x$-type and $P$-type values to 0
$P_{1,N} = 1$ //probability of "recursing" to the entire strand is 1
for $l = N, 1$ //decrement subsequence length
    Initialize $Q^x = Q^{x1}$, $Q^{x1} = Q^{x2}$, $Q^{x2} = 0$
    Initialize $P^x = P^{x1}$, $P^{x1} = P^{x2}$, $P^{x2} = 0$
    for $i = 1, N-l+1$
        $j = i+l-1$
        //$P, P^m$ recursions
        for $d = i, j-4$
            $\Delta p = P_{i,j} Q_{i,d-1} Q^s_{d,j} / Q_{i,j}$
            $P_{i,d-1} \mathrel{+}= \Delta p$
            $P^s_{d,j} \mathrel{+}= \Delta p$
            $P^{ms}_{d,j} \mathrel{+}= P^m_{i,j} \exp\{-\alpha_3[d-i]/RT\} Q^{ms}_{d,j} / Q^m_{i,j}$
            $\Delta p = P^m_{i,j} Q^m_{i,d-1} Q^{ms}_{d,j} / Q^m_{i,j}$
            $P^m_{i,d-1} \mathrel{+}= \Delta p$
            $P^{ms}_{d,j} \mathrel{+}= \Delta p$
        //$P^s, P^{ms}$ recursions
        for $d = i+4, j$ // loop over all possible rightmost pairs $i \cdot d$
            $P^b_{i,d} \mathrel{+}= P^s_{i,j} Q^b_{i,d} / Q^s_{i,j}$
            $P^b_{i,d} \mathrel{+}= P^{ms}_{i,j} Q^b_{i,d} \exp\{-[\alpha_2 + \alpha_3(j-d)]/RT\} / Q^{ms}_{i,j}$
        // $P^b$ recursion
        // Compute internal loop contributions to $P^b$ in $O(N^3)$
        call fastiloopsN3$(i, j, l, Q^b, Q^x, Q^{x2}, P^b, P^x, P^{x2})$
        for $d = i+6, j-5$
            $\Delta p = P^b_{i,j} Q^m_{i+1,d-1} Q^{ms}_{d,j-1} \exp\{-[\alpha_1 + \alpha_2]/RT\} / Q^b_{i,j}$
            $P^m_{i+1,d-1} \mathrel{+}= \Delta p$
            $P^{ms}_{d,j-1} \mathrel{+}= \Delta p$
```

Figure A1. $O(N^3)$ recursion probability algorithm that excludes pseudoknots. The algorithm proceeds from longer subsequences to shorter ones, so in contrast to the analogous partition function algorithm (see Fig. 11 of ref. 18), Q^{x1} and Q^{x2} refer to subsequences whose lengths are shorter (by 1 and 2, respectively) than the current subsequence of length l.

which relates $Q^x_{i,s}$ (for subsequences of length l) to $Q^x_{i-1,s+2}$ (for subsequences of length $l + 2$). The first line "seeds" Q^x with cases at an extension border ($L_1 = 4$ or $L_2 = 4$) for subsequent extension to longer subsequences. For conciseness, we have introduced the definition

$$\Gamma(s) \equiv \exp\{-[\gamma_1(s) + \gamma_2(|L_1 - L_2|) + \gamma_3(e, d, e+1, d-1)]/RT\} Q^b_{d,e},$$

where d and e are defined implicitly in terms of L_1 and L_2. For implementation purposes, the second line of (7) is calculated during the l, i loop and temporarily stored in $Q^{x2}_{i-1,s+2}$. The first line of (7) is added to this contribution in the $l + 2$, $i - 1$ loop. As a result of this two step procedure, we adopt the convention that L_1 and L_2 are defined with respect to the loop index in which they are calculated (i.e., l, i for the second line and $l + 2$, $i - 1$ for the first line). This convention facilitates the comparison of the extension identity with pseudocode.

The recursion probability algorithm examines subsequences of length l starting with $l = N$ and ending with $l = 1$. To recompute Q^x in this context, we use the contraction identity

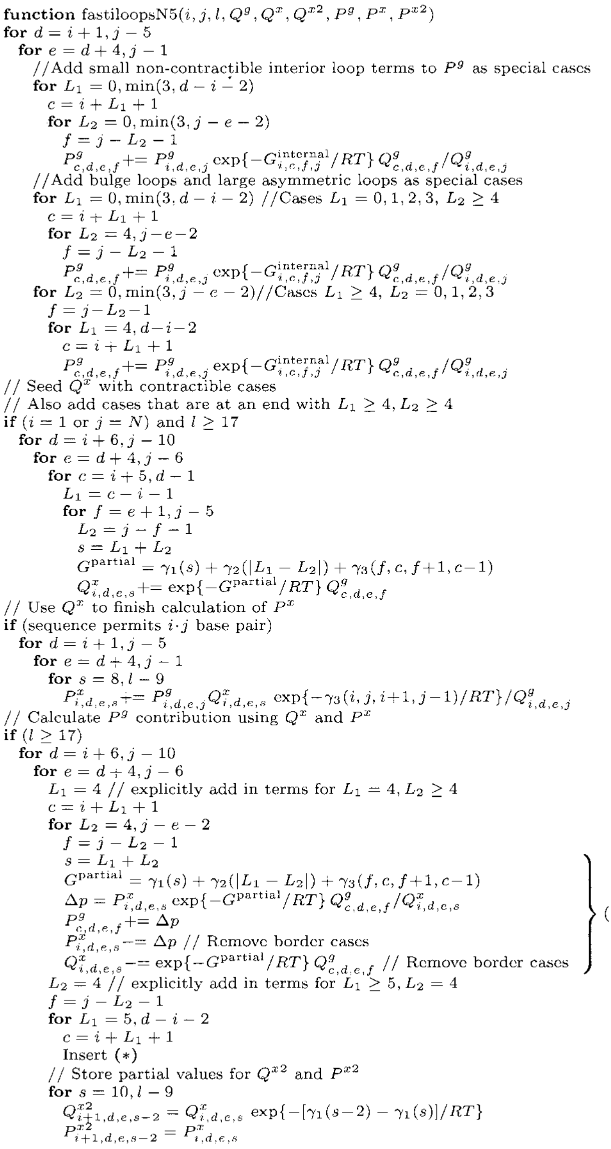
$$Q^x_{i+1,s-2} = \sum_{\substack{i=1 \\ L_1 \geq 4, L_2 \geq 4 \\ L_1+L_2=s-2}} \Gamma(s-2) + \sum_{\substack{j=N \\ L_1 \geq 4, L_2 \geq 4 \\ L_1+L_2=s-2}} \Gamma(s-2)$$
$$+ \left[\left(Q^x_{i,s} - \Gamma(s)\big|_{\substack{L_1=4 \\ L_1+L_2=s}} - \Gamma(s)\big|_{\substack{L_2=4 \\ L_1+L_2=s}}\right) \exp\{-[\gamma_1(s-2) - \gamma_1(s)]/RT\}\right] \quad (8)$$

which relates $Q^x_{i,s}$ (for subsequences of length l) to $Q^x_{i+1,s-2}$ (for subsequences of length $l - 2$). The first line "seeds" Q^x with cases that are both extensible ($L_1 \geq 4$ and $L_2 \geq 4$) and at an end of the strand ($i = 1$ or $j = N$). For implementation purposes, the second line of (8) is calculated during the l, i loop and temporarily stored in $Q^{x2}_{i+1,s-2}$. The first line of (8) is added to this contribution in the $l - 2$, $i + 1$ loop. We retain the convention that L_1 and L_2 are defined with respect to the loop index in which they are calculated (i.e., l, i for the second line and $l - 2$, $i + 1$ for the first line).

Derivation of the algorithm to compute P^x requires careful consideration. The quantities Q^x and Q^{x2} contain incomplete partition function information for "possible extensible loops," but they do not represent subsequence partition functions in the manner of other Q-type matrices. In a normal recursion relation,

```
function fastiloopsN3$(i, j, l, Q^b, Q^x, Q^{x2}, P^b, P^x, P^{x2})$
//Add small non-contractible interior loop terms to $P^b$ as special cases
for $L_1 = 0, 3$
  $d = i + L_1 + 1$
  for $L_2 = 0, \min(3, j-d-5)$
    $e = j - L_2 - 1$
    $P^b_{d,e} \mathrel{+}= P^b_{i,j} \exp\{-G^{\mathrm{internal}}_{i,d,e,j}/RT\} Q^b_{d,e} / Q^b_{i,j}$
//Add bulge loops and large asymmetric loops as special cases
for $L_1 = 0, 3$ //Cases $L_1 = 0, 1, 2, 3,\ L_2 \geq 4$
  $d = i + L_1 + 1$
  for $L_2 = 4, j-d-5$
    $e = j - L_2 - 1$
    $P^b_{d,e} \mathrel{+}= P^b_{i,j} \exp\{-G^{\mathrm{internal}}_{i,d,e,j}/RT\} Q^b_{d,e} / Q^b_{i,j}$
for $L_2 = 0, 3$//Cases $L_1 \geq 4,\ L_2 = 0, 1, 2, 3$
  $e = j - L_2 - 1$
  for $L_1 = 4, e-i-5$
    $d = i + L_1 + 1$
    $P^b_{d,e} \mathrel{+}= P^b_{i,j} \exp\{-G^{\mathrm{internal}}_{i,d,e,j}/RT\} Q^b_{d,e} / Q^b_{i,j}$
// Seed $Q^x$ with contractible cases
// Also add cases that are at an end with $L_1 \geq 4, L_2 \geq 4$
if $(i = 1$ or $j = N)$ and $l \geq 15$
  for $d = i + 5, j - 9$
    $L_1 = d - i - 1$
    for $e = d + 4, j - 5$
      $L_2 = j - e - 1$
      $s = L_1 + L_2$
      $G^{\mathrm{partial}} = \gamma_1(s) + \gamma_2(|L_1 - L_2|) + \gamma_3(e, d, e+1, d-1)$
      $Q^x_{i,s} \mathrel{+}= \exp\{-G^{\mathrm{partial}}/RT\} Q^b_{d,e}$
//Use $Q^x$ to finish calculation of $P^x$
if (sequence permits $i \cdot j$ base pair)
  for $s = 8, l - 7$
    $P^x_{i,s} \mathrel{+}= P^b_{i,j} Q^x_{i,s} \exp\{-\gamma_3(i, j, i+1, j-1)/RT\} / Q^b_{i,j}$
//Calculate $P^b$ contribution using $Q^x$ and $P^x$
if $(l \geq 15)$ // smallest subsequence not added to $P^b$ as special case
  $L_1 = 4$ // explicitly add in terms for $L_1 = 4, L_2 \geq 4$
  $d = i + L_1 + 1$
  for $L_2 = 4, j - d - 5$
    $e = j - L_2 - 1$                                                         ⎫
    $s = L_1 + L_2$                                                           ⎪
    $G^{\mathrm{partial}} = \gamma_1(s) + \gamma_2(|L_1 - L_2|) + \gamma_3(e, d, e+1, d-1)$   ⎪
    $\Delta p = P^x_{i,s} \exp\{-G^{\mathrm{partial}}/RT\} Q^b_{d,e} / Q^x_{i,s}$            ⎬ (*)
    $P^b_{d,e} \mathrel{+}= \Delta p$                                                     ⎪
    $P^x_{i,s} \mathrel{-}= \Delta p$ // Remove border cases                             ⎪
    $Q^x_{i,s} \mathrel{-}= \exp\{-G^{\mathrm{partial}}/RT\} Q^b_{d,e}$ // Remove border cases  ⎭
  $L_2 = 4$ // explicitly add in terms for $L_1 \geq 5, L_2 = 4$
  $e = j - L_2 - 1$
  for $L_1 = 5, e - i - 5$
    $d = i + L_1 + 1$
    Insert (*)
// Store partial values for $Q^{x2}$ and $P^{x2}$
for $s = 10, l - 7$
  $Q^{x2}_{i+1,s-2} = Q^x_{i,s} \exp\{-[\gamma_1(s-2) - \gamma_1(s)]/RT\}$
  $P^{x2}_{i+1,s-2} = P^x_{i,s}$
```

Figure A2. Pseudocode for computing interior loop contributions to P^b in $O(N^3)$ as an alternative to the $O(N^4)$ interior loop recursion of Figure 4.



```
function fastiloopsN5($i, j, l, Q^g, Q^x, Q^{x2}, P^g, P^x, P^{x2}$)
for $d = i + 1, j - 5$
  for $e = d + 4, j - 1$
    //Add small non-contractible interior loop terms to $P^g$ as special cases
    for $L_1 = 0, \min(3, d - i - 2)$
      $c = i + L_1 + 1$
      for $L_2 = 0, \min(3, j - e - 2)$
        $f = j - L_2 - 1$
        $P^g_{c,d,e,f} \mathrel{+}= P^g_{i,d,e,j} \exp\{-G^{\text{internal}}_{i,c,f,j}/RT\}\, Q^g_{c,d,e,f}/Q^g_{i,d,e,j}$
    //Add bulge loops and large asymmetric loops as special cases
    for $L_1 = 0, \min(3, d - i - 2)$ //Cases $L_1 = 0, 1, 2, 3,\ L_2 \geq 4$
      $c = i + L_1 + 1$
      for $L_2 = 4, j-e-2$
        $f = j - L_2 - 1$
        $P^g_{c,d,e,f} \mathrel{+}= P^g_{i,d,e,j} \exp\{-G^{\text{internal}}_{i,c,f,j}/RT\}\, Q^g_{c,d,e,f}/Q^g_{i,d,e,j}$
    for $L_2 = 0, \min(3, j - e - 2)$//Cases $L_1 \geq 4,\ L_2 = 0, 1, 2, 3$
      $f = j-L_2-1$
      for $L_1 = 4, d-i-2$
        $c = i + L_1 + 1$
        $P^g_{c,d,e,f} \mathrel{+}= P^g_{i,d,e,j} \exp\{-G^{\text{internal}}_{i,c,f,j}/RT\}\, Q^g_{c,d,e,f}/Q^g_{i,d,e,j}$
// Seed $Q^x$ with contractible cases
// Also add cases that are at an end with $L_1 \geq 4, L_2 \geq 4$
if ($i = 1$ or $j = N$) and $l \geq 17$
  for $d = i + 6, j - 10$
    for $e = d + 4, j - 6$
      for $c = i + 5, d - 1$
        $L_1 = c - i - 1$
        for $f = e + 1, j - 5$
          $L_2 = j - f - 1$
          $s = L_1 + L_2$
          $G^{\text{partial}} = \gamma_1(s) + \gamma_2(|L_1 - L_2|) + \gamma_3(f, c, f{+}1, c{-}1)$
          $Q^x_{i,d,e,s} \mathrel{+}= \exp\{-G^{\text{partial}}/RT\}\, Q^g_{c,d,e,f}$
// Use $Q^x$ to finish calculation of $P^x$
if (sequence permits $i \cdot j$ base pair)
  for $d = i + 1, j - 5$
    for $e = d + 4, j - 1$
      for $s = 8, l - 9$
        $P^x_{i,d,e,s} \mathrel{+}= P^g_{i,d,e,j} Q^x_{i,d,e,s} \exp\{-\gamma_3(i, j, i{+}1, j{-}1)/RT\}/Q^g_{i,d,e,j}$
// Calculate $P^g$ contribution using $Q^x$ and $P^x$
if ($l \geq 17$)
  for $d = i + 6, j - 10$
    for $e = d + 4, j - 6$
      $L_1 = 4$ // explicitly add in terms for $L_1 = 4, L_2 \geq 4$
      $c = i + L_1 + 1$
      for $L_2 = 4, j - e - 2$
        $f = j - L_2 - 1$
        $s = L_1 + L_2$
        $G^{\text{partial}} = \gamma_1(s) + \gamma_2(|L_1 - L_2|) + \gamma_3(f, c, f{+}1, c{-}1)$
        $\Delta p = P^x_{i,d,e,s} \exp\{-G^{\text{partial}}/RT\}\, Q^g_{c,d,e,f}/Q^x_{i,d,e,s}$
        $P^g_{c,d,e,f} \mathrel{+}= \Delta p$                                    } (*)
        $P^x_{i,d,e,s} \mathrel{-}= \Delta p$ // Remove border cases
        $Q^x_{i,d,e,s} \mathrel{-}= \exp\{-G^{\text{partial}}/RT\}\, Q^g_{c,d,e,f}$ // Remove border cases
      $L_2 = 4$ // explicitly add in terms for $L_1 \geq 5, L_2 = 4$
      $f = j - L_2 - 1$
      for $L_1 = 5, d - i - 2$
        $c = i + L_1 + 1$
        Insert (*)
      // Store partial values for $Q^{x2}$ and $P^{x2}$
      for $s = 10, l - 9$
        $Q^{x2}_{i+1,d,e,s-2} = Q^x_{i,d,e,s} \exp\{-[\gamma_1(s{-}2) - \gamma_1(s)]/RT\}$
        $P^{x2}_{i+1,d,e,s-2} = P^x_{i,d,e,s}$
```

Figure A3. Pseudocode for computing interior loop contributions to P^g in $O(N^5)$ as an alternative to the $O(N^6)$ interior loop recursion of Figure 5.

Q-type matrices on the right-hand side are subsequence partition functions describing a local structural motif that contributes to the larger subsequence partition function on the left-hand side. $Q^x_{i,s}$ contains information about possible extensible loops that may not actually exist (if i and j are not complementary). The extension identity (7) passes this potentially useful information on to $Q^{x2}_{i-1,s+2}$. Consider, for example, a chain of Q^x values related by the extension identity in a case where no complementary $i \cdot j$ base pair is encountered while incrementing l by 2 until an end of the strand is reached. In this scenario, the values of Q^x computed in this chain should not contribute to the corresponding recursion probabilities P^x because the values of Q^x are not identified with any secondary structure in the equilibrium ensemble. Hence, the calculation of P^x requires information about which Q^x quantities ultimately contribute to secondary structures in the ensemble. As a result, the extension identity (7) cannot simply be transformed using the standard recursion probability approach, which assumes that both sides of the equation represent subsequence partition functions that are assured of contributing to the equilibrium ensemble. This realization suggests computing $P^x_{i,s}$ by adding the probabilities of all internal loops that rely on $Q^x_{i,s}$ to incorporate information in the partition function.

To calculate $P^x_{i,s}$ (for a fixed l), note that $Q^x_{i,s}$ will be invoked for all interior loops (i', d, e, j') with interior pair $d \cdot e$ and closing pair $i' \cdot j'$ such that

$$i - i' = j' - j \geq 0, \quad L_1 \geq 4, \quad L_2 \geq 4, \quad L_1 + L_2 = s, \qquad (9)$$

where L_1, L_2 and s are defined with respect to i and j. Hence, a particular loop (i', d, e, j') is identified with a set of $Q^x_{i,s}$ terms that are related by the extension identity (7). Alternatively, a particular $Q^x_{i,s}$ term is identified with all of the interior loops (i', d, e, j') to which it ultimately contributes via the extension identity. Consequently, from the notion of recursion probabilities introduced earlier, $P^x_{i,s}$ (for a fixed l) should be the sum of the probabilities of all interior loops (i', d, e, j') that satisfy the properties (9). For the case where $i - 1 \leq N - j$ (the case $i - 1 > N - j$ yields analogous results), it follows that

$$P^x_{i,s} = \sum_{i'=1}^{i} \sum_{\substack{L_1 \geq 4, L_2 \geq 4 \\ L_1 + L_2 = s}} p(i', d, e, j'), \qquad (10)$$

where $p(i', d, e, j')$ is the probability of the (i', d, e, j') interior loop in the equilibrium ensemble of secondary structures. Because $P^{x2}_{i+1,s-2}$ is defined similarly, with l and s decremented by 2, it follows that

$$P^{x2}_{i+1,s-2} = \sum_{i'=1}^{i+1} \sum_{\substack{L_1 \geq 5, L_2 \geq 5 \\ L_1 + L_2 = s}} p(i', d, e, j'), \qquad (11)$$

where L_1 and L_2 are temporarily defined with respect to i and j to retain the size constraint $L_1 + L_2 = s$. Comparing (10) and (11), we then identify the relationship

$$P^{x2}_{i+1,s-2} = \sum_{\substack{L_1 \geq 5, L_2 \geq 5 \\ L_1 + L_2 = s}} p(i', d, e, j')|_{i'=i+1, j'=j-1}$$

$$+ \left[P^x_{i,s} - \sum_{i'=1}^{i} p(i', d, e, j')|_{\substack{L_1 = 4, L_2 \geq 4 \\ L_1 + L_2 = s}} - \sum_{i'=1}^{i} p(i', d, e, j')|_{\substack{L_1 \geq 5, L_2 = 4 \\ L_1 + L_2 = s}} \right],$$

where L_1 and L_2 continue to be defined with respect to i and j. Finally, we shift the indices in the first line so that L_1 and L_2 are defined with respect to $i + 1$ and $j - 1$

$$P^{x2}_{i+1,s-2} = \sum_{\substack{L_1 \geq 4, L_2 \geq 4 \\ L_1 + L_2 = s-2}} p(i', d, e, j')\big|_{i'=i+1, j'=j-1} + \left[P^x_{i,s} - \sum_{i'=1}^{i} p(i', d, e, j')\Big|^{L_1=4, L_2 \geq 4}_{L_1+L_2=s} - \sum_{i'=1}^{i} p(i', d, e, j')\Big|^{L_1 \geq 5, L_2 = 4}_{L_1+L_2=s} \right]. \quad (12)$$

This identity relates $P^x_{i,s}$ (for subsequences of length l) to $P^x_{i+1,s-2}$ (for subsequences of length $l - 2$). For implementation purposes, the second line is calculated during the l, i loop and temporarily stored in $Q^{x2}_{i+1,s-2}$. Each of the sums of form $\sum_{i'=1}^{i}$ operates on a single term, which is a subset of the terms in the definition of $P^x_{i,s}$ (10). Hence, the sums of form $\sum_{i'=1}^{i}$ in (12) may be evaluated implicitly as $P^x_{i,s}$ times a quotient with $Q^x_{i,s}$ in the denominator and the corresponding subset of $Q^x_{i,s}$ in the numerator. The first line is added to this contribution in the $l - 2$, $i + 1$ loop. There, the summation corresponds to exactly those loops treated by $Q^x_{i+1,s-2}$ in the case where $i + 1$ and $j - 1$ base pair. As usual, L_1 and L_2 are defined with respect to the loop index in which they are calculated (i.e., l, i for the second line and $l - 2$, $i + 1$ for the first line).

References

1. Tinoco, I., Jr.; Uhlenbec, O.; Levine, M. Nature 1971, 230, 362.
2. Turner, D. H.; Sugimoto, N.; Freier, S. Annu Rev Biophys Biophys Chem 1988, 17, 167.
3. SantaLucia, J., Jr. Proc Natl Acad Sci USA 1998, 95, 1460.
4. Mathews, D.; Sabina, J.; Zuker, M.; Turner, D. J Mol Biol 1999, 288, 911.
5. Zuker, M. Curr Opin Struct Biol 2000, 10, 303.
6. Waterman, M. In Studies in Foundations and Combinatorics: Advances in Mathematics Supplemental Studies; Academic Press: New York, 1978, 1, 167.
7. Waterman, M.; Smith, T. Math Biosci 1978, 42, 257.
8. Nussinov, R.; Pieczenik, J.; Griggs, J.; Kleitman, D. SIAM J Appl Math 1978, 35, 68.
9. Zuker, M.; Stiegler, P. Nucleic Acids Res 1981, 9, 133.
10. Hofacker, I.; Fontana, W.; Stadler, P.; Bonhoeffer, L.; Tacker, M.; Schuster, P. Chem Monthly 1994, 125, 167.
11. McCaskill, J. Biopolymers 1990, 29, 1105.
12. Bonhoeffer, S.; McCaskill, J.; Stadler, P.; Schuster, P. Eur Biophys J 1993, 22, 13.
13. van Batenburg, F.; Gultyaev, A.; Pleij, C.; Ng, J. Nucleic Acids Res 2000, 28, 201.
14. Winfree, E.; Liu, F.; Wenzler, L.; Seeman, N. C. Nature 1998, 394, 539.
15. Yan, H.; LaBean, T.; Feng, L.; Reif, J. Proc Natl Acad Sci USA 2003, 100, 8103.
16. Rivas, E.; Eddy, S. J Mol Biol 1999, 285, 2053.
17. Akutsu, T. Discrete Appl Math 2000, 104, 45.
18. Dirks, R.; Pierce, N. A. J Comput Chem 2003, 24, 1664.
19. Comolli, L.; Smirnov, I.; Xu, L.; Blackburn, E.; James, T. Proc Natl Acad Sci USA 2002, 99, 16998.
20. Theimer, C.; Finger, L.; Trantirek, L.; Feigon, J. Proc Natl Acad Sci USA 2003, 100, 449.
21. Lyngso, R.; Zuker, M.; Pedersen, C. Bioinformatics 1999, 15, 440.
22. Ding, Y.; Lawrence, C. Nucleic Acids Res 2003, 31, 7280.
23. Dirks, R.; Lin, M.; Winfree, E.; Pierce, N. A. Nucleic Acids Res 2004, 32, 1392.
24. Fu, T.-J.; Seeman, N. C. Biochemistry 1993, 32, 3211.
25. Seeman, N. C. J Theor Biol 1982, 99, 237.