

A dynamic programming algorithm for RNA structure prediction including pseudoknots

Elena Rivas and Sean R. Eddy¹

*Department of Genetics,
Washington University, St. Louis, MO 63130 USA*

Abstract

We describe a dynamic programming algorithm for predicting optimal RNA secondary structure, including pseudoknots. The algorithm has a worst case complexity of $\mathcal{O}(N^6)$ in time and $\mathcal{O}(N^4)$ in storage. The description of the algorithm is complex, which led us to adopt a useful graphical representation (Feynman diagrams) borrowed from quantum field theory. We present an implementation of the algorithm that generates the optimal minimum energy structure for a single RNA sequence, using standard RNA folding thermodynamic parameters augmented by a few parameters describing the thermodynamic stability of pseudoknots. We demonstrate the properties of the algorithm by using it to predict structures for several small pseudoknotted and non-pseudoknotted RNAs. Although the time and memory demands of the algorithm are steep, we believe this is the first algorithm to be able to fold optimal (minimum energy) pseudoknotted RNAs with the accepted RNA thermodynamic model.

Running title RNA pseudoknot prediction by dynamic programming.

Keywords RNA, secondary structure prediction, pseudoknots, dynamic programming, thermodynamic stability.

¹To whom correspondence should be addressed. Tel: +1 314 362 7666; Fax: +1 314 362 7855; Email: eddy@genetics.wustl.edu.

1 INTRODUCTION

Many RNAs fold into structures that are important for regulatory, catalytic, or structural roles in the cell. An RNA's structure is dominated by base pairing interactions, most of which are Watson-Crick pairs between complementary bases. The base paired structure of an RNA is called its secondary structure. Because Watson-Crick pairs are such a stereotyped and relatively simple interaction, accurate RNA secondary structure prediction appears to be an achievable goal.

A rather reliable approach for RNA structure prediction is comparative sequence analysis, in which covarying residues (e.g. compensatory mutations) are identified in a multiple sequence alignment of RNAs with similar structures but different sequences (Woese & Pace, 1993). Covarying residues, particularly pairs which covary to maintain Watson-Crick complementarity, are indicative of conserved base pairing interactions. The accepted secondary structures of most structural and catalytic RNAs were generated by comparative sequence analysis.

If one has only a single RNA sequence (or a small family of RNAs with little sequence diversity) comparative sequence analysis cannot be applied. Here the best current approaches are energy minimization algorithms (Schuster *et al.*, 1997). While not as accurate as comparative sequence analysis, these algorithms have still proven to be useful research tools. Thermodynamic parameters are available for predicting the ΔG of a given RNA structure (Freier *et al.*, 1986; Serra *et al.*, 1995). The Zuker algorithm (implemented in the programs MFOLD (Zuker, 1989a) and ViennaRNA (Schuster *et al.*, 1994)) is an efficient dynamic programming algorithm for identifying the globally minimal energy structure for a sequence, as defined by such a thermodynamic model (Zuker & Stiegler, 1981; Zuker & Sankoff, 1984; Sankoff, 1985). The Zuker algorithm requires $O(N^3)$ time and $O(N^2)$ space for a sequence of length N , and so is reasonably efficient and practical even for large RNA sequences. The Zuker dynamic programming algorithm was subsequently extended to allow experimental constraints, and to sample suboptimal folds (Zuker, 1989b). McCaskill's variant of the Zuker algorithm calculates probabilities (confidence estimates) for particular base pairs (McCaskill, 1990).

One well-known limitation of the Zuker algorithm is that it is incapable of

predicting so-called *RNA pseudoknots*. This is the problem that we address in this paper.

The thermodynamic model for non-pseudoknotted RNA secondary structure includes some stereotypical interactions, such as stacked base-paired stems, hairpins, bulges, internal loops, and multiloops. Formally, non pseudoknotted structures obey a “nesting” convention: that for any two base pairs i, j and k, l (where $i < j$ and $k < l$), either $i < k < l < j$ or $k < i < j < l$. It is precisely this “nesting” convention that the Zuker dynamic programming algorithm relies upon to recursively calculate the minimal energy structure on progressively longer subsequences. An RNA pseudoknot is defined as a structure containing base pairs which violate the nesting convention. An example of a simple pseudoknot is shown in Figure 1.

RNA pseudoknots are functionally important in several known RNAs (ten Dam *et al.*, 1992). For example, by comparative analysis, RNA pseudoknots are conserved in ribosomal RNAs, the catalytic core of group I introns, and RNase P RNAs. Plausible pseudoknotted structures have been proposed (Florentz *et al.*, 1982), and recently confirmed (Kolk *et al.*, 1998) for the 3' end of several plant viral RNAs, where pseudoknots are apparently used to mimic tRNA structure. In vitro RNA evolution (SELEX) experiments have yielded families of RNA structures which appear to share a common pseudoknotted structure, such as RNA ligands selected to bind HIV-1 reverse transcriptase (Tuerk *et al.*, 1992).

Most methods for RNA folding which are capable of folding pseudoknots adopt heuristic search procedures and sacrifice optimality. Examples of these approaches include quasi-Monte Carlo searches (Abrahams *et al.*, 1990) and genetic algorithms (Gulyaev *et al.*, 1995; van Batenburg *et al.*, 1995). These approaches are inherently unable to guarantee that they have found the “best” structure given the thermodynamic model, and consequently unable to say how far a given prediction is from optimality.

A different approach to pseudoknot prediction based on the maximum weighted matching (MWM) algorithm (Edmonds, 1965; Gabow, 1976) was introduced by Cary and Stormo (1995). Using the MWM algorithm, an optimal structure is found, even in the presence of complicated knotted interactions, in $O(N^3)$ time and $O(N^2)$ space. However, MWM seems best suited to folding sequences

for which a previous alignment exists. In the scoring system used by Cary and Stormo, weights are assigned by comparative analysis. It is not clear to us that the MWM algorithm will be amenable to folding single sequences or collections of sequences which present little variation respect to each other. However, we believe that this was the first work that indicated that optimal RNA pseudoknot predictions can be made with polynomial time algorithms. It had been widely believed, but never proven, that pseudoknot prediction would be an NP problem (NP = nondeterministic polynomial; e.g. only solvable by heuristic or brute force approaches).

In this paper we describe a dynamic programming algorithm which finds optimal pseudoknotted RNA structures. We describe the algorithm using a diagrammatic representation borrowed from quantum field theory (Feynman diagrams). We implement a version of the algorithm that finds minimal energy RNA structures using the standard RNA secondary structure thermodynamic model (Freier *et al.*, 1986, Serra *et al.*, 1995), augmented by a few pseudoknot-specific parameters that are not yet available in the standard folding parameters, and by coaxial stacking energies (Walter *et al.*, 1994) for both pseudoknotted and non-pseudoknotted structures. We demonstrate the properties of the algorithm by testing it on several small RNA structures, including both structures thought to contain pseudoknots and structures thought not to contain pseudoknots.

2 ALGORITHM

In this section we will introduce a diagrammatic way of representing RNA folding algorithms. We will start by describing the Nussinov algorithm (Nussinov *et al.*, 1978), and the Zuker-Sankoff algorithm (Zuker & Sankoff 1984; Sankoff 1985) in the context of this representation. Later on we will extend the diagrammatic representation to include pseudoknots and coaxial stackings. The Nussinov and Zuker-Sankoff algorithms can be implemented without the diagrammatic representation, but this representation is essential to manage the complexity introduced by pseudoknots.

2.1 Preliminaries

From here on, unless otherwise stated, a flat solid line will represent the backbone of a RNA sequence with its 5' end placed in the left hand side of the segment. N will represent the length (in number of nucleotides) of the RNA.

Secondary interactions will be represented by wavy lines connecting the two interacting positions in the backbone chain, while the backbone itself always remains flat. No more than two bases are allowed to interact at once. This representation does not provide insight about real (3D) spatial arrangements, but is very convenient for algorithmic purposes. When necessary for clarification single stranded regions will be marked by dots, but when unambiguous, dots will be omitted for simplicity. Using this representation (figure 2) we can describe hairpins, bulges, stems, internal loops and multiloops as simple nested structures; a pseudoknot, on the other hand, corresponds to a non-nested structure.

2.2 Diagrammatic representation of nested algorithms

In order to describe a nested algorithm we need to introduce two triangular $N \times N$ matrices, to be called vx and wx . These matrices are defined in the following way: $vx(i, j)$ is the score of the best folding between positions i and j provided that i and j are paired to each other; whereas $wx(i, j)$ is the score of the best folding between positions i and j regardless of whether i and j pair to each other or not. These matrices are graphically represented in the form indicated in fig. 3. The filled inner space indicates that we do not know how many interactions (if any) occur for the nucleotides inside, in contrast with a blank inner space which indicates that the fragment inside is known to be single stranded. The wavy line in vx , as previously, indicates that i and j are definitely paired, and similarly the discontinuous line in wx indicates that the relation between i and j is unknown. Also part of our convention is that for a given fragment, nucleotide i is at the 5'-end, and nucleotide j is at the 3'-end, so that $i \leq j$.

The purpose of a dynamic programming algorithm is to fill the vx and wx matrices with appropriate numerical weights by means of some sort of recursive calculation.

The recursion relations used to fill the wx matrix include: single-stranded nucleotides, external pairs, external dangling bases, and bifurcations. The actual recursion is easier to understand by looking at the diagrams involved (given in fig. 4) and the recursion can be expressed as,

$$wx(i, j) = \text{optimal} \left\{ \begin{array}{l} P + vx(i, j) \\ L_{i+1, j-1}^i + R_{i+1, j-1}^j + P + vx(i+1, j-1) \\ L_{i+1, j}^i + P + vx(i+1, j) \\ R_{i, j-1}^j + P + vx(i, j-1) \\ Q + wx(i+1, j) \\ Q + wx(i, j-1) \\ wx(i, k) + wx(k+1, j) \quad [\forall k, \quad i \leq k \leq j]. \end{array} \right. \begin{array}{l} \text{paired} \\ \text{dangles} \\ \text{single} \\ \text{stranded} \\ \text{bifurcations} \end{array} \quad (1)$$

Each line gives the formal score of one of the diagrams in fig. 4. The diagram on the left is calculated as the score of the best diagram on the right. Here P is some value that represents the score for a base pair. Q represents the score for a single stranded nucleotide, whereas $L_{i+1, j}^i$ and $R_{i, j-1}^j$ stand for the score of a nucleotide dangling off a base pair of nucleotides, at the 5'-end or the 3'-end respectively.

The recursion for wx includes hairpins, bulges, internal loops, and multiloops. But what is special about hairpins, bulges, internal loops, and multiloops in this diagrammatic representation? To answer this question we have to introduce two more definitions: *Surfaces* (S) and *Irreducible Surfaces* (IS).

Roughly speaking a Surface is any alternating sequence of solid and wavy lines that closes on itself. An Irreducible Surface is a Surface such that if one of the H-bonds (or secondary interactions) is broken there is no other surface contained inside, that is, an IS cannot be “reduced” to any other surface. (ISs are similar to the “k-loops” defined by Sankoff (1985).) The *order*, \mathcal{O} , of an IS is given by the number of wavy lines (secondary interactions), which is equal to the number of solid-line intervals. It is easy to see that hairpin loops constitute the ISs of $\mathcal{O}(1)$; stems, bulges and internal loops are all the ISs of $\mathcal{O}(2)$, and what are referred to in the literature as “multiloops” are the ISs of $\mathcal{O} > 2$.

The actual recursion for vx is given in fig. 5, and can be expressed as,

$$vx(i, j) = \mathit{optimal} \left\{ \begin{array}{l} IS^1(i, j) \\ IS^2(i, j : k, l) + vx(k, l) \\ IS^3(i, j : k, l : m, n) + vx(k, l) + vx(m, n) \\ IS^4(i, j : k, l : m, n : r, s) + vx(k, l) \\ \quad + vx(m, n) + vx(r, s) \\ \mathcal{O}(IS^5) \end{array} \right. \quad (2)$$

$$[\forall k, l, m, n, r, s, \quad i \leq k \leq l \leq r \leq s \leq m \leq n \leq j]$$

This recursion is an expansion in ISs of successively higher order. Here $IS^n(i_1, j_1 : i_2, j_2 : \dots : i_n, j_n)$ represent the score for an IS of order n , in which i_k is paired to j_k . This general algorithm is quite impractical, because each IS of $\mathcal{O}(\gamma)$ adds a complexity of $\mathcal{O}(N^{2\gamma})$ to the calculation. [An IS of $\mathcal{O}(N^{2\gamma})$ requires us to search through 2γ independent segments in the entire sequence of N nucleotides.] To make it useful we have to truncate the expansion in IS's at some order in the recursion for vx in fig. 5. The symbol $\mathcal{O}(IS^5)$ indicates the order of IS at which we truncate the recursion. (Note that the recursion for wx will remain always the same.)

These recursions are equivalent to those proposed by Sankoff (1985) in Theorem 2. Notice also that in defining the recursive algorithm we have not yet had to specify anything about the particular manner in which the contribution from different IS's are calculated in order to obtain the most optimal folding.

The simplest truncation is to stop at order zero. In this approximation none of the ISs (hairpin, bulge, internal loop...) are given any specialized scores. We only have to provide a specific score for a base pair, B . The recursion for vx then simplifies to fig. 6, and can be cast into the form,

$$vx(i, j) = B + wx(i + 1, j - 1). \quad (3)$$

If we set $B = P = +1$, and $Q = 0$ in equation (1) then we have the Nussinov algorithm (Nussinov *et al.*, 1978). This simple algorithm calculates the folding with the maximum number of base pairs.

The next order of complexity we explore corresponds to a truncation at ISs of $\mathcal{O}(2)$. Hairpin loops, bulges, stems, and internal loops are treated with precision

by the scoring functions IS^1 and IS^2 . The rest of ISs, collected under the name of “multiloops”—which are much less frequent than the previous—are described in an approximate form. The diagrams of this approximation are given in fig. 7, and correspond to,

$$vx(i, j) = \mathit{optimal} \begin{cases} IS^1(i, j) \\ IS^2(i, j : k, l) + vx(k, l) \\ P_I + M + wx_I(i + 1, k) + wx_I(k + 1, j - 1) \end{cases} \quad (4)$$

$$[\forall k, l \quad i \leq k \leq l \leq j]$$

This is the algorithm described by Sankoff (1985) in theorem 3. This is the approximation that MFOLD (Zuker, 1981) and *ViennaRNA* (Schuster *et al.*, 1994) implement. P_I stands for the scoring parameter for a pair in a multiloop, and parameter M stands for the score for a multiloop. Note that the matrix wx_I used to truncate the recursion for vx in (4) does not have to be the same as the one used in (1). Matrix wx_I is used exclusively for diagrams which will be incorporated into multiloops. Although both matrices wx and wx_I have similar recursions, the parameters of these two recursions will have in general different values (represent here by P, Q, L, R and P_I, Q_I, L_I, R_I respectively). This feature is implemented both in MFOLD and in our program.

Higher orders of specificity of the general algorithm are possible, but are certainly more time consuming, and they have not been explored so far. One reason for this relative lack of development is that there is little information about the energetic properties of multiloops. The generalized nested algorithm provides a way to unify the currently available dynamic algorithms for RNA folding. At a given order, the error of the approximation is given by the difference between the assigned score to “multiloops” and the precise score that one of those higher order ISs deserves.

2.3 Description of the pseudoknot algorithm

Pseudoknots are non-nested configurations and clearly cannot be described with just the wx and vx matrices we introduced in the previous section. The key point of the pseudoknot algorithm is the use of *gap matrices* in addition to the wx

and vx matrices. Looking at the graphical representation of one of the simplest pseudoknots, fig. 8, we can see that we could describe such a configuration by putting together two gap matrices with complementary holes.

The pseudoknot dynamic programming algorithm uses one-hole or gap matrices (fig. 9) as a generalization of the wx and vx matrices. Let us define $whx(i, j : k, l)$ as the graph that describes the best folding that connects segments $[i, k]$ with $[l, j]$, $i \leq k \leq l \leq j$ such that the relation between i and j and k and l is undetermined. Similarly we define $vhx(i, j : k, l)$ as the graph that describes the best folding that connects segments $[i, k]$ with $[l, j]$, $i \leq k \leq l \leq j$ such that i and j are base-paired and k and l are also base-paired. For completeness we have to introduce also matrix $yhx(i, j : k, l)$ in which k and l are paired, but the relation between i and j is undetermined, and its counterpart $zhx(i, j : k, l)$ in which i and j are paired, but the relation between k and l is undetermined.

The non-gap matrices wx , vx are contained as a particular case of the gap matrices. When there is no hole, $k = l - 1$, then by construction,

$$\begin{aligned} whx(i, j : k, k + 1) &= wx(i, j) \\ zhx(i, j : k, k + 1) &= vx(i, j) \quad \forall k, \quad i \leq k \leq j. \end{aligned} \tag{5}$$

We have the gap matrices as the building blocks of the algorithm, but how do we establish a consistent and complete recursion relation? Here is where the analogy between the gap matrices and the Feynman diagrams of quantum field theory was of great help (Bjorken & Drell 1965).²

Let us start with the generalization of the recursions for wx and vx in the presence of gap matrices. A non-gap matrix can be obtained by combining two gap matrices together, therefore the recursions for wx and vx add one more diagram with two gap matrices to recursions (1) and (2). Again the diagrammatic representation (fig. 10, 11) is more helpful than words in explaining the recursion. Note that the new term introduced in both recursions involves two gap matrices. In fact the recursion is an expansion in the number of gap matrices necessary at each step of the recursion.

²More precisely, the analogy is more cleanly expressed in terms of Schwinger-Dyson diagrams which in QFT are used to represent full interacting vertices and propagators recursively in terms of elementary interactions.

$$wx(i, j) = \text{optimal} \left\{ \begin{array}{l} P + vx(i, j) \\ L_{i+1, j-1}^i + R_{i+1, j-1}^j + P + vx(i+1, j-1) \\ L_{i+1, j}^i + P + vx(i+1, j) \\ R_{i, j-1}^j + P + vx(i, j-1) \\ Q + wx(i+1, j) \\ Q + wx(i, j-i) \\ wx(i, k) + wx(k+1, j) \\ G_w + whx(i, r : k, l) + whx(k+1, j : l-1, r+1) \\ \mathcal{O}(whx + whx + whx) \end{array} \right. \quad (6)$$

Where G_w denotes the score for introducing a pseudoknot.

Similarly for vx ,

$$vx(i, j) = \text{optimal} \left\{ \begin{array}{l} IS^1(i, j) \\ IS^2(i, j : k, l) + vx(k, l) \\ P_I + M + wx_I(i+1, k) + wx_I(k+1, j-1) \\ P_I + \widetilde{M} + G_{wI} + whx(i+1, r : k, l) \\ \quad + whx(k+1, j-1 : l-1, r+1) \\ \mathcal{O}(whx + whx + whx) \end{array} \right. \quad (7)$$

$$[\forall i, k, l, r, j \quad i \leq k \leq l \leq r \leq j]$$

Here \widetilde{M} stands for a generic score for generating a non-nested multiloop, and G_{wI} stands for the score for generating an internal pseudoknot.

Practical considerations make us truncate the expansion at this point, so we will not include diagrams that require three or more gap matrices. This statement should not mislead one into thinking that we cannot deal with complicated pseudoknots. The recursive nature of the approximation allows us to describe overlapping pseudoknots (defined as those pseudoknots for which a planar representation does not require crossing lines) as well as non-planar pseudoknots (for which a planar representation requires crossing lines). The *Escherichia coli* α mRNA presented by Gluick *et al.* (1994) is an example of a non-planar RNA pseudoknot that can be parsed using the pseudoknot algorithm. However the algorithm is not able to find all possible knotted configurations (fig. 12). Nevertheless, the approximation seems to be adequate for the currently known pseudoknots in RNA folding.

Note that two approximations are involved in the algorithm. Apart from that just mentioned (how to truncate the infinite expansion in gap matrices to make the algorithm polynomial), we also use the approximation previously introduced for the nested algorithm (that IS's of $\mathcal{O} > 2$ or multiloops are described in some approximated form).

The algorithm is not complete until we provide the full recursive expressions to calculate the gap matrices. For a given gap matrix, we have to consider in how many different ways that diagram can be assembled using one or two matrices at a time. The full description of those diagrams is given in Subsection 5.2 . (Again Feynman diagrams are of great use here.)

2.4 Coaxial stacking

It is quite frequent in RNA folding to create a more stable configuration when two independent configurations stack coaxially. That occurs for instance, when two hairpin loops with their respective stems are contiguous, so that one of them can fall on top of the other creating a more stable configuration than when the two hairpins just coexist without interaction of any kind.

The algorithm implements coaxial energies for both nested and non-nested structures. We adopt the coaxial energies provided by Walter *et al.* (1994) for coaxial stacking of nested structures. For coaxial stacking of non-nested structures we multiply these previous energies by an estimated (ad hoc) weighting parameter $g < 1$.

Using our diagrammatic representation it is possible to be systematic in describing the possible coaxial stacking that can occur. In the general recursion one has to look for contiguous nucleotides and allow them to be explicitly paired—but not to each other. This is best understood with an example. Consider the recursion for wx in fig. 10, in particular the bifurcation diagram

$$wx(i, j) \longrightarrow wx(i, k) + wx(k + 1, j), \quad \forall k, i \leq k \leq j.$$

In order to allow for the possibility of coaxial stacking such diagram has to be complemented with another one in which the nucleotides of the bifurcation are base-paired

$$wx(i, j) \longrightarrow vx(i, k) + vx(k + 1, j) + C(k, i : k + 1, j), \quad \forall k, i \leq k \leq j.$$

This new diagram indicates that if nucleotides k and $k + 1$ are paired to nucleotides i and j respectively, that configuration is specially favored by an amount $C(k, i : k + 1, j)$ (presumably negative in energy units) because both sub-structures, $vx(i, k)$ and $vx(k + 1, j)$, will stack onto each other.

Notice also that the new diagram really corresponds to four new diagrams because once we allow pairing, dangling bases have also to be considered, so the full nearest-neighbour interaction is taken into account. For purposes of clarity we will not explicitly specify any of the extra diagrams with dangling involved. The rest of the additional diagrams to be included in the recursions to take care of coaxial stackings are also given in Subsection 5.2 along with the full set of diagrams. Coaxial diagrams can be recognized by the empty dots representing the contiguous coaxially-stacking nucleotides.

2.5 Minimum-energy implementation. Thermodynamic parameters

We have implemented the pseudoknot algorithm using thermodynamic parameters in order to fill the scoring matrices, both gapped and ungapped. For the relevant nested structures: hairpin loops, bulges, stems, internal loops and multiloops we have used the same set of energies as used in MFOLD.³ Free energies for coaxial stacking C were obtained from Walter *et al.*, (1994). Table 2 provides a list of the parameters used for nested conformations.

For the non-nested configurations, there is not much thermodynamic information available (Wyatt *et al.*, 1990; Gluick *et al.*, 1994). This is not an untypical situation; there is already very little thermodynamic information available for regular multiloops, let alone for pseudoknots. We had to tune by hand the parameters related to pseudoknots. For some non-nested structures we multiplied the nested parameters by an estimated weighting parameter $g < 1$. It would be very useful, in order to improve the accuracy of this thermodynamic implementation of the pseudoknot algorithm, to have more accurate experimentally-based determinations of these parameters. Table 3 provides a list of the parameters

³Since the implementation of the pseudoknot algorithm the Turner group has produced a new complete and more accurate list of parameters (Matthews *et al.*, 1998) which we have not yet implemented.

we used for pseudoknot-related conformations.

3 RESULTS

The main purpose of this paper is to present an algorithm that solves optimal pseudoknotted RNA structures by dynamic programming. RNA structure prediction of single sequences with nested algorithms already involves some approximation and inaccuracy (Zuker, 1995; Huynen *et al*, 1997). We expect this inaccuracy to increase in our case since the algorithm now allows a much larger configuration space. Therefore our limited objective here is to show that on a few small RNAs that are thought to conserve pseudoknots, our program (a minimal-energy implementation of the pseudoknot algorithm using a thermodynamic model) will actually find the pseudoknots; and for a few small RNAs that do not conserve pseudoknots, our program finds results similar to MFOLD, and does not introduce spurious pseudoknots.

3.1 tRNAs

Almost all transfer RNAs (tRNA) share a common cloverleaf structure. We have tested the algorithm on a group of 25 tRNAs selected at random from the Sprinzl tRNA database (Steinberg *et al.*, 1993). The program finds no spurious pseudoknot for any of the tested sequences. All tRNAs fold into a cloverleaf configuration but one (DT5090). Of the 24 cloverleaf foldings, 15 are completely consistent with their proposed structures (that is, each helical region has at least 3 base pairs in common with its proposed folding). The remaining 9 cloverleaf foldings misplace one (6 sequences) or two (3 sequences) of the helical regions. On the other hand, MFOLD's lowest-energy prediction for the same set of tRNA sequences includes only 19 cloverleaf foldings of which 14 are completely consistent with their proposed structures. Performance for our program is therefore at least comparable to MFOLD; the inaccuracies found are the result of the approximations in the thermodynamic model, not a problem with the pseudoknot algorithm per se. The relevant result in relation to the pseudoknot algorithm is that its implementation predicts no spurious pseudoknots for tRNAs.

One should not think of this result as a trivial one, because when knots

are allowed, the configuration space available becomes much larger than the observed class of conformations. This problem is particularly relevant for “maximum-pairing-like” algorithms, such as the MWM algorithm presented by Cary & Stormo (1995) or a Nussinov implementation of our pseudoknot algorithm (fig 6). In both cases, the result is almost universal pairing because there is enough freedom to be able to coordinate any position with another one in the sequence.

Another important aspect of tRNA folding is coaxial energies. Most tRNAs gain stability by stacking coaxially two of the hairpin loops, and the third one with the acceptor stem. This aspect of tRNA folding is very important and in some cases crucial to determine the right structure. There are situations like tRNA DA0260 in which MFOLD does not assign the lowest energy to the correct structure (the MFOLD 3.0 prediction for DA0260 misses the acceptor stem, and has a free energy of -22.0 Kcal/mol). Our algorithm, on the other hand, implements coaxial energies; as a result the cloverleaf configuration becomes the most stable folding for tRNA DA0260 ($\Delta G = -24.3$ Kcal/mol). The implementation of coaxial energies explains why we found more cloverleaf structures for tRNAs than MFOLD does.

3.2 HIV-1-RT-ligand RNA pseudoknots

High-affinity ligands of the reverse transcriptase of human immunodeficiency virus type 1 (HIV-1) isolated by a SELEX procedure by Tuerk *et al.* (1992) seem to have a pseudoknot consensus secondary structure. These oligonucleotides have between 34 and 47 bases and fold into a simple pseudoknot. Of a total of 63 SELEX-selected pseudoknotted sequences available from Tuerk *et al.* (1992), we found 54 foldings that agreed exactly with the structures derived by comparative analysis. ($\Delta G = -10.9$ Kcal/mol for sequence pattern I (3-2)). As expected, MFOLD predicts only one of the two stems ($\Delta G = -7.5$ Kcal/mol for the same sequence).

3.3 Viral RNAs

Some virus RNA genomes [such as turnip yellow mosaic virus (TYMV) (Guilley *et al.*, 1979)] present a tRNA-like structure at their 3' end that includes a

pseudoknot in the aminoacyl acceptor arm very close to the 3' end (Kolk *et al.*, 1998; Florentz *et al.*, 1982; Dumas *et al.*, 1987). Our program predicts correctly the TYMV tRNA-like structure with its pseudoknot for the last 86 bases at the 3' end with $\Delta G = -30.4$ Kcal/mol (the MFOLD 3.0 prediction for TYMV has a free energy of $\Delta G = -28.9$ Kcal/mol). The tRNA-like 3' terminal structure is conserved among tymoviruses and also for the tobacco mosaic virus cowpea strain (CcTMV) another valine acceptor. Of the seven valine-acceptor tRNA-like structures proposed to date (Van Belkum *et al.*, 1987) we reproduce six of them, except for kennedya yellow mosaic virus (KYMV).

Finally we have considered the last 189 bases of the 3' terminal of the Tobacco mosaic virus (TMV) (Van Belkum *et al.*, 1985). TMV also has a tRNA-like structure at the end, but it may have additional upstream pseudoknots, up to a total of five, forming a long quasi-continuous helix. We folded the upstream and downstream regions separately in a piece of 84 nucleotides (the folding requires 52 minutes and 9.8 Mb) and 105 nucleotides (the folding requires 246 minutes and 22.5 Mb) respectively. Our program predicts the 105 nucleotides downstream region exactly with $\Delta G = -32.5$ Kcal/mol. For the 84 nucleotides upstream region we find four of the five helical regions with $\Delta G = -19.0$ Kcal/mol.

4 DISCUSSION

In this paper we present an algorithm able to predict pseudoknots by dynamic programming. This algorithm demonstrates that using certain approximations consistent with the accepted Turner thermodynamic model, the prediction of pseudoknotted structures is a problem of polynomial complexity (although admittedly high). Having an optimal dynamic programming algorithm will enable extending other dynamic programming based methods that rigorously explore the conformational space for RNA folding (McCaskill, 1990; Bonhoeffer *et al.*, 1993) to pseudoknotted structures.

Apart from the usefulness of the algorithm in predicting pseudoknots, we also include coaxial energies (when two stems stack coaxially), a very common feature of RNA folding. We expect MFOLD will also include coaxial energies in the near future (Matthews *et al.*, 1998).

Our algorithm is presented in the context of a general framework in which a generic folding is expressed in terms of its elementary secondary interactions (which we have identified as the irreducible surfaces). This is a further generalization of Sankoff's result (1985). The calculation of an optimal folding becomes an expansion in ISs of increasingly higher order. Our formalization incorporates all current dynamic programming RNA folding algorithms in addition to our pseudoknot algorithm. It also establishes the limitations of each approximation by determining at which order the expansion is truncated.

As for the thermodynamic implementation presented in this paper, one of our major problems is the almost complete lack of thermodynamic information about pseudoknot configurations. The thermodynamic algorithm is also sensitive to the accuracy of the existing thermodynamic parameters. We expect to improve this aspect by implementing the more complete set of parameters provided by the Turner group (Matthews *et al.*, 1998).

The principal drawback is the time and memory constraints imposed by the computational complexity of the algorithm. At this early stage, we cannot analyze sequences much larger than 130-140 bases. For now the program is adequate for folding small RNAs. A 100 nucleotide RNA takes about 4 hours and 22.5 MB to fold on an SGI R10K Origin200.

Due to practical limitations, at a given point in the recursion we only allow the incorporation of two gap matrices. However, since each of those gap matrices can in turn be assembled by other two of those matrices, it implies that the algorithm includes in its configuration space a large variety of knotted motifs. The limitations of this truncation appeared when we considered applying this approach to describe pairwise residue interactions in protein folding. A parallel β -sheet configuration provides an example of a complicated knotted folding that cannot be handled by the pseudoknot algorithm presented in this paper. However, all known RNA pseudoknots can be handled by the algorithm, which makes the approximation useful enough for RNA secondary structure.

Although we implemented the algorithm for energy minimization, extending MFOLD to pseudoknotted structures, the algorithm is not limited to energy minimization. Our algorithm can be converted into a probabilistic model for pseudoknot-containing RNA folding. Probabilistic models of RNA secondary

structure based on “stochastic context free grammar” (SCFG) formalisms (Eddy *et al.*, 1994; Sakakibara *et al.*, 1994; Lefebvre, 1996) have been introduced both for RNA single-sequence folding and for RNA structural alignment and structural similarity searches. The Inside and CYK dynamic programming algorithms used for SCFG-based structural alignment are fundamentally similar to the Zuker algorithm (Durbin *et al.*, 1998), and have consequently also been unable to deal with pseudoknots. Heuristic approaches to applying SCFG-like structural alignment models to pseudoknots have been introduced (Brown, 1996; Notredame *et al.*, 1997). An SCFG-like probabilistic version of our pseudoknot algorithm could be designed to obtain optimal structural alignment of pseudoknot-containing RNAs.

5 METHODS

5.1 Implementation

The algorithm was implemented on a Silicon Graphics Origin200. The algorithm has a theoretical worst-case complexity of $\mathcal{O}(N^6)$ in time and $\mathcal{O}(N^4)$ in storage. At its present stage, the program is empirically observed to run $\mathcal{O}(N^{6.8})$ in time and $\mathcal{O}(N^{3.8})$ in memory. For instance, a tRNA of 75 nucleotides takes 24 minutes and uses 6.6 Mb of memory. The 3' end of tobacco mosaic virus has 105 nucleotides and takes 246 minutes and uses 22.5 Mb. The program empirically scales above the theoretical complexity in time of the algorithm. This effect seems to have to do with the way the machine allocates memory for larger RNAs. The software and parameter sets are available by request from E. Rivas (elena@genetics.wustl.edu).

5.2 Complete set of diagrams for the pseudoknot algorithm

In this section we provide the complete recursion relations for all the matrices used in the pseudoknot algorithm.

The recursion for the non-gap matrix wx is given by (cf. fig. 13):

$$wx(i, j) = \text{optimal} \left\{ \begin{array}{l} P + vx(i, j) \\ L_{i+1, j-1}^i + R_{i+1, j-1}^j + P + vx(i+1, j-1) \\ L_{i+1, j}^i + P + vx(i+1, j) \\ R_{i, j-1}^j + P + vx(i, j-1) \\ Q + wx(i+1, j) \\ Q + wx(i, j-1) \\ wx(i, k) + wx(k+1, j) \\ C(k, i : k+1, j) + vx(i, k) + vx(k+1, j) \\ G_w + whx(i, r : k, l) + whx(k+1, j : l-1, r+1) \\ 2 * P + G_w + C(l-1, r+1 : l, k) \\ \quad + yhx(i, r : k, l) + yhx(k+1, j : l-1, r+1) \end{array} \right. \begin{array}{l} \text{paired} \\ \text{dangles} \\ \text{single} \\ \text{stranded} \\ \text{nested} \\ \text{bifurcations} \\ \text{non-nested} \\ \text{bifurcations} \end{array} \quad (8)$$

Here G_w denotes a score for introducing a pseudoknot, and $C(l, r : l+1, k)$ is a special score for a coaxial stacking of pairs (l, r) and $(l+1, k)$.

We should also remember that the algorithm uses two different wx matrices depending on whether the subset $i\dots j$ is free-standing (wx) or appears inside a multiloop (in which case we use wx_I). Both recursions are identical apart from having different coefficients as described in table 2.

The recursion for the non-gap matrix vx is given by (cf. fig. 14):

$$\begin{array}{l}
\left. \begin{array}{l}
IS^1(i, j) \\
IS^2(i, j : k, l) + vx(k, l) \\
P_I + M + wx_I(i + 1, k) + wx_I(k + 1, j - 1) \\
2 * P_I + C(i, j : i + 1, k) + M + vx(i + 1, k) + wx_I(k + 1, j - 1) \\
2 * P_I + C(j - 1, k + 1 : j, i) + M + wx_I(i + 1, k) + vx(k + 1, j - 1) \\
3 * P_I + C(k, i' : k + 1, j') + M + vx(i', k) + vx(k + 1, j') \\
P_I + \widetilde{M} + G_{wI} + whx(i + 1, r : k, l) \\
\quad + whx(k + 1, j - 1 : l - 1, r + 1) \\
2 * P_I + \widetilde{M} + G_{wI} + C(i, j, : i + 1, r) \\
\quad + zhx(i + 1, r : k, l) + whx(k + 1, j - 1 : l - 1, r + 1) \\
2 * P_I + \widetilde{M} + G_{wI} + C(j - 1, k + 1 : j, i) \\
\quad + whx(i + 1, r : k, l) + zhx(k + 1, j - 1 : l - 1, r + 1) \\
3 * P_I + \widetilde{M} + G_{wI} + C(l - 1, r + 1 : l, k) \\
\quad + yhx(i + 1, r : k, l) + yhx(k + 1, j - 1 : l - 1, r + 1)
\end{array} \right\} vx(i, j) = optimal \quad \left. \begin{array}{l}
 \\
 \\
 \\
 \\
 \\
 \\
\phantom{P_I + \widetilde{M} + G_{wI} + whx(i + 1, r : k, l)} \\
\phantom{P_I + \widetilde{M} + G_{wI} + whx(i + 1, r : k, l)} \\
\phantom{2 * P_I + \widetilde{M} + G_{wI} + C(i, j, : i + 1, r)} \\
\phantom{2 * P_I + \widetilde{M} + G_{wI} + C(i, j, : i + 1, r)} \\
\phantom{2 * P_I + \widetilde{M} + G_{wI} + C(j - 1, k + 1 : j, i)} \\
\phantom{2 * P_I + \widetilde{M} + G_{wI} + C(j - 1, k + 1 : j, i)} \\
\phantom{3 * P_I + \widetilde{M} + G_{wI} + C(l - 1, r + 1 : l, k)} \\
\phantom{3 * P_I + \widetilde{M} + G_{wI} + C(l - 1, r + 1 : l, k)}
\end{array} \right\} \begin{array}{l}
\text{IS(1)} \\
\text{IS(2)} \\
\text{nested} \\
\text{multiloops} \\
\text{non-nested} \\
\text{multiloops}
\end{array}
\end{array} \tag{9}$$

$$[\forall i, i', k, l, r, j', j \quad i \leq i' \leq k \leq l \leq r \leq j' \leq j]$$

Parameters P_I , \widetilde{M} , and G_{wI} are defined in table 2.

The initialization conditions are

$$wx(i, i) = 0, \tag{10}$$

$$vx(i, i) = +\infty.$$

$$[\forall i \quad 1 \leq i \leq N]$$

The recursion for the vhx matrix in the pseudoknot algorithm is given by (cf. fig. 15):

$$vx(i, j : k, l) = optimal \left\{ \begin{array}{l}
\widetilde{IS}^2(i, j : k, l) \\
\widetilde{IS}^2(i, j : r, s) + vhx(r, s : k, l) \\
\widetilde{IS}^2(r, s : k, l) + vhx(i, j : r, s) \\
2 * \widetilde{P} + \widetilde{M} + whx(i + 1, j - 1 : k - 1, l + 1)
\end{array} \right. \tag{11}$$

$$[\forall i, r, k, l, s, j \quad i \leq r \leq k \leq l \leq s \leq j]$$

Here \widetilde{M} correspond to the score given to a multiloop in a vhx gap matrix. It could be equal to M , the score defined for multiloops in the vx matrix, but it does not have to be. Similarly, the score for an irreducible surface of $\mathcal{O}(2)$, \widetilde{IS}^2 , could be the same as the one given for nested structures, IS^2 , but again, it does not have to be. We found the best fits by giving them values different to the ones used for nested foldings (cf. table 2 and table 3).

The recursions for the gap matrices zhx and yhx in the pseudoknot algorithm are complementary and given by (cf. fig. 16 and fig. 17):

$$\begin{array}{l}
zhx(i, j : k, l) = \textit{optimal} \left\{ \begin{array}{l}
\left[\begin{array}{l}
\widetilde{P} + vhx(i, j : k, l) \\
\widetilde{L} + \widetilde{R} + \widetilde{P} + vhx(i, j : k - 1, l + 1) \\
\widetilde{R} + \widetilde{P} + vhx(i, j : k - 1, l) \\
\widetilde{L} + \widetilde{P} + vhx(i, j : k, l + 1)
\end{array} \right] \begin{array}{l}
\text{paired} \\
\text{dangles}
\end{array} \\
\left[\begin{array}{l}
\widetilde{Q} + zhx(i, j : k - 1, l) \\
\widetilde{Q} + zhx(i, j : k, l + 1)
\end{array} \right] \begin{array}{l}
\text{single} \\
\text{stranded}
\end{array} \\
\left[\begin{array}{l}
zhx(i, j : r, l) + wx_I(r + 1, k) \\
2 * \widetilde{P} + C(r, l : r + 1, k) + vhx(i, j : r, l) + vx(r + 1, k) \\
zhx(i, j : k, s) + wx_I(l, s - 1) \\
2 * \widetilde{P} + C(s - 1, l : s, k) + vhx(i, j : k, s) + vx(l, s - 1) \\
\widetilde{IS}^2(i, j : r, s) + zhx(r, s : k, l) \\
\widetilde{P} + \widetilde{M} + whx(i + 1, j - 1 : k, l)
\end{array} \right] \begin{array}{l}
\text{nested} \\
\text{bifurcations}
\end{array}
\end{array} \right.
\end{array}
\tag{12}$$

$$\begin{array}{l}
\left. \begin{array}{l}
\tilde{P} + vhx(i, j : k, l) \\
\tilde{L} + \tilde{R} + \tilde{P} + vhx(i + 1, j - 1 : k, l) \\
\tilde{L} + \tilde{P} + vhx(i + 1, j : k, l) \\
\tilde{R} + \tilde{P} + vhx(i, j - 1 : k, l) \\
\tilde{Q} + yhx(i + 1, j : k, l) \\
\tilde{Q} + yhx(i, j - 1 : k, l) \\
wx_I(i, r) + yhx(r + 1, j : k, l) \\
2 * \tilde{P} + C(r, i : r + 1, j) + vx(i, r) + vhx(r + 1, j : k, l) \\
yhx(i, s : k, l) + wx_I(s + 1, j) \\
2 * \tilde{P} + C(s, i : s + 1, j) + vhx(i, s : k, l) + vx(s + 1, j) \\
yhx(i, j : r, s) + \widetilde{IS}^2(r, s : k, l) \\
\tilde{P} + \widetilde{M} + whx(i, j : k - 1, l + 1)
\end{array} \right\} \begin{array}{l}
\text{paired} \\
\text{dangles} \\
\text{single} \\
\text{stranded} \\
\text{nested} \\
\text{bifurcations}
\end{array}
\end{array}$$

(13)

$$[\forall i, r, k, l, s, j \quad i \leq r \leq k \leq l \leq s \leq j]$$

Finally, the recursion for the gap matrix whx appears in fig. 18, and is given

by:

$$\begin{array}{l}
whx(i, j : k, l) = \text{optimal} \left\{ \begin{array}{l}
2 * \tilde{P} + vhx(i, j : k, l) \\
\tilde{P} + zhx(i, j : k, l) \\
\tilde{P} + yhx(i, j : k, l) \\
\tilde{L} + \tilde{R} + 2 * \tilde{P} + vhx(i + 1, j : k - 1, l) \\
\tilde{L} + \tilde{R} + 2 * \tilde{P} + vhx(i, j - 1 : k, l + 1) \\
2 * \tilde{L} + 2 * \tilde{P} + vhx(i + 1, j : k, l + 1) \\
2 * \tilde{R} + 2 * \tilde{P} + vhx(i, j - 1 : k - 1, l) \\
\tilde{L} + 2 * \tilde{R} + 2 * \tilde{P} + vhx(i + 1, j - 1 : k - 1, l) \\
2 * \tilde{L} + \tilde{R} + 2 * \tilde{P} + vhx(i + 1, j : k - 1, l + 1) \\
2 * \tilde{L} + \tilde{R} + 2 * \tilde{P} + vhx(i + 1, j - 1 : k, l + 1) \\
\tilde{L} + 2 * \tilde{R} + 2 * \tilde{P} + vhx(i, j - 1 : k - 1, l + 1) \\
2 * \tilde{L} + 2 * \tilde{R} + 2 * \tilde{P} + vhx(i + 1, j - 1 : k - 1, l + 1) \\
\tilde{L} + \tilde{R} + \tilde{P} + zhx(i + 1, j - 1 : k, l) \\
\tilde{L} + \tilde{P} + zhx(i + 1, j : k, l) \\
\tilde{R} + \tilde{P} + zhx(i, j - 1 : k, l) \\
\tilde{L} + \tilde{R} + \tilde{P} + yhx(i, j : k - 1, l + 1) \\
\tilde{R} + \tilde{P} + yhx(i, j : k - 1, l) \\
\tilde{L} + \tilde{P} + yhx(i, j : k, l + 1) \\
\tilde{Q} + whx(i + 1, j : k, l) \\
\tilde{Q} + whx(i, j - 1 : k, l) \\
\tilde{Q} + whx(i, j : k - 1, l) \\
\tilde{Q} + whx(i, j : k, l + 1)
\end{array} \right. \begin{array}{l}
\text{paired} \\
\text{dangles} \\
\text{single} \\
\text{stranded}
\end{array}
\end{array}$$

(14)

(cont.)

$$zhx(i, j : k, k) = zhx(i, j : k, k + 1) = vx(i, j).$$

$$[\forall i, k, j \quad 1 \leq i \leq k \leq j \leq N]$$

Acknowledgments

This work was supported by NIH grant HG01363 and by a gift from Eli Lilly. The idea for the algorithm came from a discussion with Gary Stormo at a meeting at the Aspen Center for Physics. Tim Hubbard suggested parallel β -strands in proteins as an example of a set of pairwise interactions that the algorithm cannot handle.

References

- Abrahams, J.P., van der Berg, M., van Batenburg, E. & Pleij, C.W.A. (1990). Prediction of RNA secondary structure, including pseudoknotting, by computer simulation. *Nucl. Acids Res.* **18**, 3035–44.
- Ahlquist, P., Dasgupta, R. & Kaesberg, P. (1981). Near identity of 3' RNA secondary structure in bromoviruses and cucumber mosaic virus. *Cell* **23**, 183-9.
- Bjorken, J.D. & Drell, S.D. (1965). *Relativistic Quantum Fields*, McGraw-Hill, New York, NY.
- Bonhoeffer, S., McCaskill, J.S., Stadler, P.F. & Schuster, P. (1993). Statistics of RNA secondary structure. *Eur. Biophys. J. (EHU)* **22**, 13–24.
- Brown, M. (1996). RNA pseudoknot modeling using intersections of stochastic context free grammars with applications to database search. *Pacific Symposium on Biocomputing 1996*.
- Cary, R.B. & Stormo, G.D. (1995). Graph-theoretic approach to RNA modeling using comparative data. *ISBM-95*, Eds.: C. Rawlings & others. AAAI Press. 75–80.
- Chomsky, D. (1959). On certain formal properties of grammars. *Information and Control* **2**, 137–76.
- Dumas, P., Moras, D., Florentz, C., Giegé, R., Verlaan, P., van Belkum, A. & Pleij, C.W.A. (1987). 3-D graphics modeling of the tRNA-like 3' end of turnip yellow mosaic virus RNA: structural and functional implications. *J. Biomol. Struct. Dyn.* **4**, 707-28.
- Durbin, R., Eddy, S.R., Krogh, A. & Mitchison, G.J. (1998). *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, Cambridge UK.
- Eddy, S.R. & Durbin, R. (1994). RNA sequence analysis using covariance models. *Nucl. Acids Res.* **22**, 2079–88.
- Edmonds, J. (1965). Maximum matching and polyhedron with 0,1-vertices. *J. Res. Nat. Bur. Stand.* **69B**, 125-30.
- Florentz, C., Briand, J.P., Romboy, P., Hirth, L., Ebel, J.P. & Giege, R. (1982). The tRNA-like structure of turnip yellow mosaic virus RNA: structural organization of the last 159 nucleotides from the 3' OH terminus. *EMBO J.* **1**, 269-76.
- Freier, S., Kierzek, R., Jaeger, J.A., Sugimoto, N., Caruthers, M.H., Neilson, T. &

- Turner, D.H. (1986). Improved free-energy parameters for predictions of RNA duplex stability. *Proc. Natl. Acad. Sci. USA*, **83**, 9373–7.
- Gabow, H.N. (1976). An efficient implementation of Edmonds' algorithm for maximum matching on graphs. *J. Asc. Com. Mach.* **23**, 221-34.
- Gluick, T.C. & Draper, D.E. (1994). Thermodynamics of folding a pseudoknotted mRNA fragment. *J. Mol. Biol.* **241**, 246-262.
- Guilley, H., Jonard, G., Kukla, B. & Richards, K.E. (1979). Sequence of 1000 nucleotides at the 3' end of tobacco mosaic virus RNA. *Nucl. Acids Res.* **6**, 1287-308.
- Gulyaev A.P., van Batenburg F.H. & Pleij C.W.A. (1995). The computer simulation of RNA folding pathways using a genetic algorithm. *J. Mol. Biol.* **250**, 37-51.
- Huynen, M., Gutell, R., & Konings, D. (1997). Assessing the reliability of RNA folding using statistical mechanics. *J. Mol. Biol.* **267**, 1104-12.
- Kolk, M.H., van der Graff, M., Wijmenga, S.S., Pleij, C.W.A., Heus, H.A. & Hilbers, C.W. (1998). NMR structure of a classical pseudoknot: interplay of single- and double-stranded RNA. *Science* **280**, 434-8.
- Lefebvre, F. (1996). A grammar-based unification of several alignments and folding algorithms. *ISBM-96*, Eds.: C. Rawlings & others. AAAI Press. 143–54.
- Matthews, D.H., Andre, T.C., Kim, J., Turner, D.H. & Zuker, M. (1998). An updated recursive algorithm for RNA secondary structure prediction with improved free energy parameters. *Molecular Modeling of Nucleic Acids*. Eds.: N. B. Leontis & J. SantaLucia Jr. American Chemical Society.
- McCaskill J.S (1990). The equilibrium partition function and base pair bindings probabilities for RNA secondary structure. *Biopolymers (A5Z)* **29**, 1105–19.
- Notredame, C., O'Brien, E.A., & Higgins, D.G. (1997). RAGA: RNA sequence alignment by genetic algorithm. *Nucl. Acids Res.* **25**, 4570–80.
- Nussinov, R., Pieczenik, G., Griggs, J.R., & Kleitman, D.J. (1978). Algorithms for loop matchings. *SIAM J. Appl. Math.* **35**, 68–82.
- Sakakibara, Y., Brown, M., Hughey, R., Mian, I.S., Sjölander, K., Underwood, R.C. & Hussler, D. (1994). Stochastic context-free grammars for tRNA modeling. *Nucl. Acids Res.* **22**, 5112–20.
- Sankoff, D. (1985). Simultaneous solution of the RNA folding alignment and pro-

- tosequence problems. *SIAM J. Appl. Math.* **45**, 810–25.
- Schuster, P., Fontana, W., Stadler, P.F. & Hofacker, I.L. (1994). From sequences to shapes and back: a case study in RNA secondary structure. *Proc. R. Soc. Lond. B. Biol. Sci.* **255**, 279–84. <http://www.itc.univie.ac.at/ivo/RNA>
- Schuster, P., Fontana, W., Stadler, P.F. & Renner, A. (1997). RNA structures and folding: from conventional to new issues in structure predictions. *Curr. Opin. Struct. Biol.* **7**, 229–35.
- Serra, M.J. & Turner, D.H. (1995). Predicting the thermodynamic properties of RNA. *Meth. Enzymol.* **259**, 242–61.
- Steinberg, S., Misch, A. & Sprinzl, M. (1993). Compilation of RNA sequences and sequences of tRNA genes. *Nucl. Acids Res.* **21**, 3011–15.
- ten Dam E., Pleij, K. & Draper, D. (1992). Structural and functional aspects of RNA pseudoknots. *Biochemistry* **31**, 11665-11676.
- Tuerk, C., MacDougal, S. & Gold, L. (1992). RNA pseudoknots that inhibit human immunodeficiency virus type 1 reverse transcriptase. *Proc. Natl. Acad. Sci. USA*, **89**, 6988–92.
- van Batenburg, F.H.D., Gulyaev, A.P. & Pleij, C.W.A. (1995). An APL-programmed genetic algorithm for the prediction of RNA secondary structure. *J. Theor. Biol.* **174**, 269–80.
- Van Belkum, A., Abrahams, J.P., Pleij, C.W.A. & Bosch, L. (1985). Five pseudoknots are present at the 204 nucleotides long 3' non coding region of tobacco mosaic virus RNA. *Nucl. Acids Res.* **13**, 7673–86.
- Van Belkum, A., Bingkun, J., Pleij, C.W.A. & Bosch, L. (1987). Structural similarities among valine-accepting tRNA-like structures in tymoviral RNAs and elongator tRNAs. *Biochemistry* **26**, 1144-51.
- Walter, A., Turner, D., Kim, J., Lyttle, M., Müller, P., Matthews, D. & Zuker, M. (1994). Coaxial stacking of helices enhances binding of oligoribonucleotides and improves predictions of RNA folding. *Proc. Natl. Acad. Sci. USA*, **91**, 9218–22.
- Woese C.R. & Pace N.R. (1993). Probing RNA structure, function, and history by comparative analysis. *The RNA World*. Eds.: R. F. Gesteland & J. F. Atkins. Cold Spring Harbor Laboratory Press. New York NY. 91-117.
- Wyatt, J.R., Puglisi, J.D. & Tinoco, I.Jr. (1990). RNA pseudoknots: stability and

- loop size requirements. *J. Mol. Biol.* **214**, 455-70.
- Zuker, M. & Stiegler, P. (1981). Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucl. Acids Res.* **9**, 133-48.
- Zuker, M. & Sankoff, D. (1984). RNA secondary structure and their prediction. *Bull. Math. Biol.* **46**, 591-621.
- Zuker, M. (1989). Computer prediction of RNA structure. *Meth. Enzymol.* **180**, 262-88.
- Zuker, M. (1989). On finding all suboptimal foldings of an RNA molecule. *Science* **244**, 48-52.
- Zuker, M. (1995). "Well-determined" regions in RNA secondary structure prediction: analysis of small subunit ribosomal RNA. *Nucl. Acids Res.* **23**, 2791-8.

TABLES

matrix $(i \leq k \leq l \leq j)$	relationship i, j	relationship k, l
$wx(i, j)$	undetermined	—
$vx(i, j)$	paired	—
$whx(i, j : k, l)$	undetermined	undetermined
$vhx(i, j : k, l)$	paired	paired
$zhx(i, j : k, l)$	paired	undetermined
$yhx(i, j : k, l)$	undetermined	paired

Table 1: Specifications of the matrices used in the pseudoknot algorithm.

Symbol	Scoring parameter for	Value(Kcal/mol)
IS^1	hairpin loops	varies
IS^2	bulges, stems and int loops	varies
C	coaxial stacking	varies
P	external pair	0
Q	single stranded base	0
R, L	base dangling off an external pair	$dangle + Q$
P_I	pair in a multiloop	0.1
Q_I	not paired base inside multiloop	0.4
R_I, L_I	base dangling off a multiloop pair	$dangle + Q_I$
M	nested multiloop	4.6

Table 2: This table includes all the parameters for which there is thermodynamic information provided by the Turner group. This parameters are identical to those used in MFOLD (<http://www.ibc.wustl.edu/~zucker/rna>).

Symbol	Scoring parameter for	Value(Kcal/mol)
\widetilde{IS}^2	IS^2 in a gap matrix	$IS^2 * g(0.83)$
\widetilde{C}	coaxial stacking in pseudoknots	$C * g$
\widetilde{P}	pair in a pseudoknot	0.1
\widetilde{Q}	not paired base in pseudoknot	0.2
$\widetilde{R}, \widetilde{L}$	base dangling off a pseudoknot pair	$dangle * g + \widetilde{Q}$
\widetilde{M}	non-nested multiloop	8.43
G_w	generating a new pseudoknot	7.0
G_{wI}	generating a pseudoknot in a multiloop	13.0
G_{wh}	overlapping pseudoknots	6.0

Table 3: In this table we introduce the new thermodynamic parameters specific for pseudoknot configurations what we had to estimate.

FIGURES

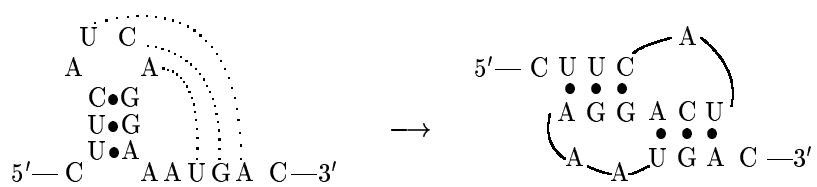


Figure 1: A simple pseudoknot. In a pseudoknot, nucleotides inside a hairpin loop pair with nucleotides outside the stem-loop.

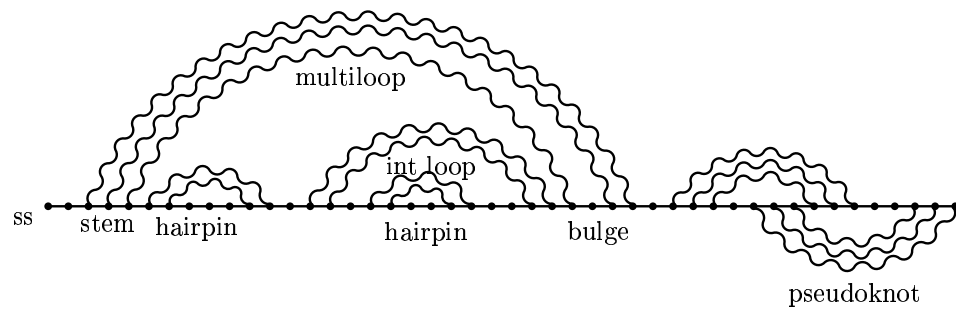


Figure 2: Diagrammatic representation of RNA most relevant secondary structures, including a pseudoknot.

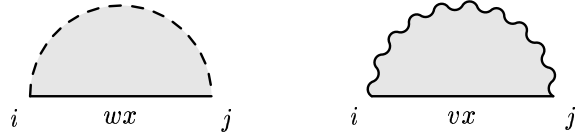


Figure 3: Wx and vx matrices.

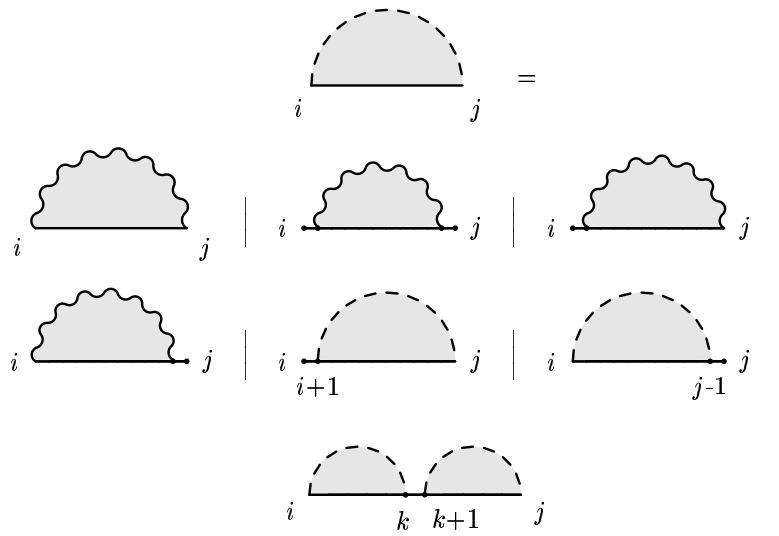


Figure 4: Recursion for wx in the nested algorithm

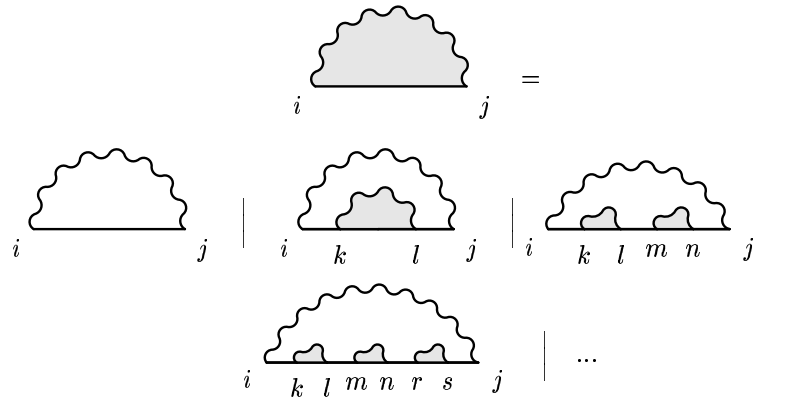


Figure 5: General recursion for vx in the nested algorithm.

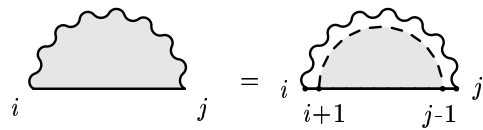


Figure 6: Recursion for vx truncated at $\mathcal{O}(0)$

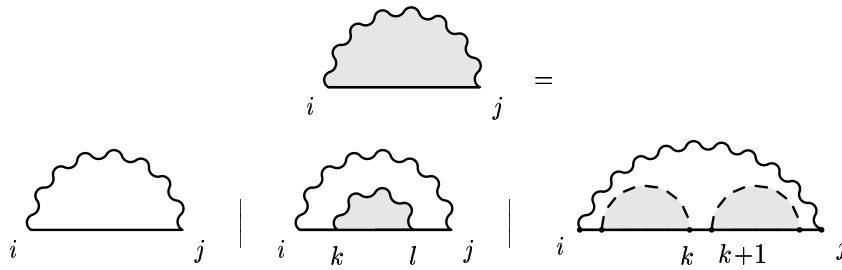


Figure 7: Recursion for vx truncated at $\mathcal{O}(2)$.

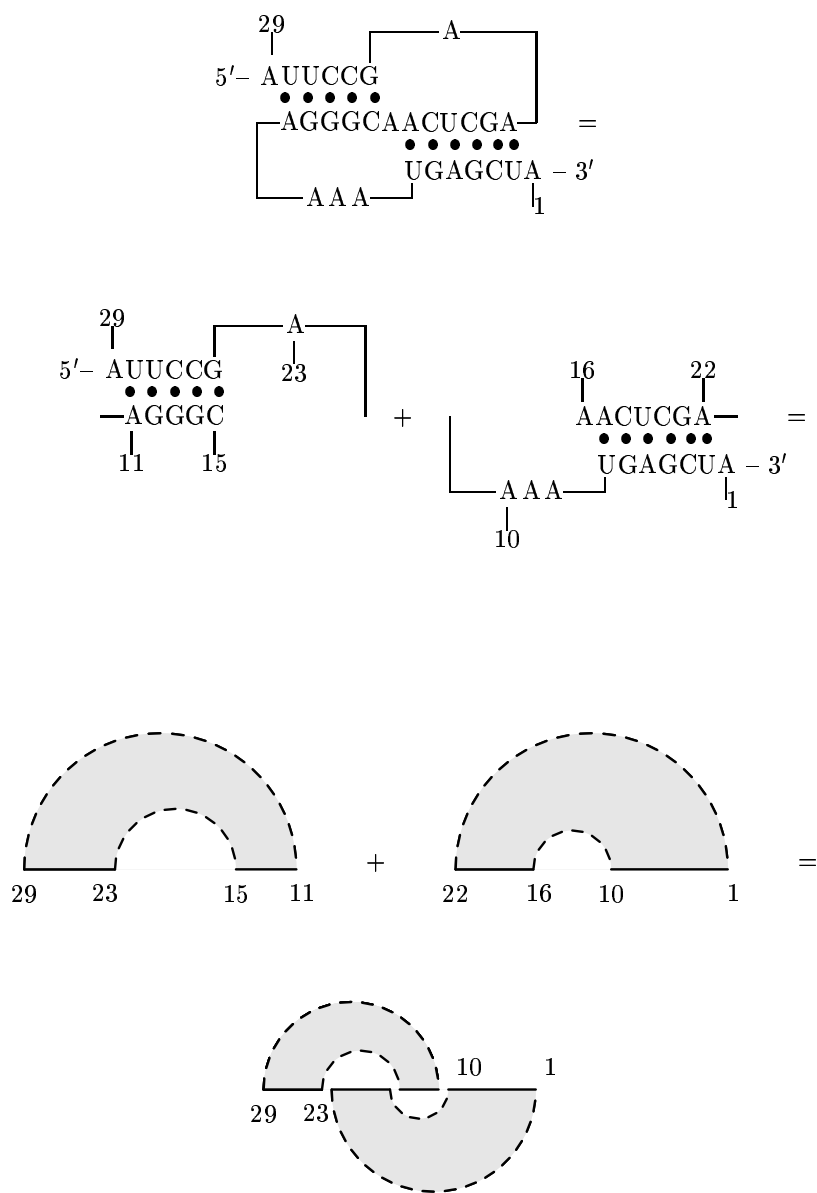


Figure 8: Construction of a simple pseudoknot using two gap matrices.

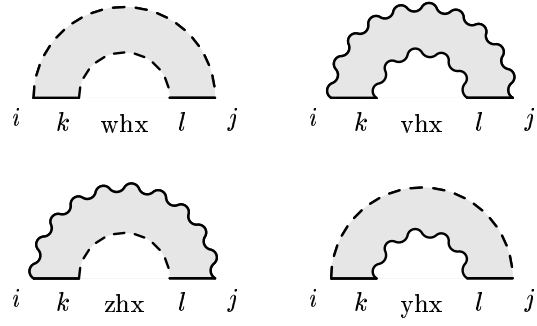


Figure 9: Representation of the gap matrices used in the algorithm for pseudo-knots.

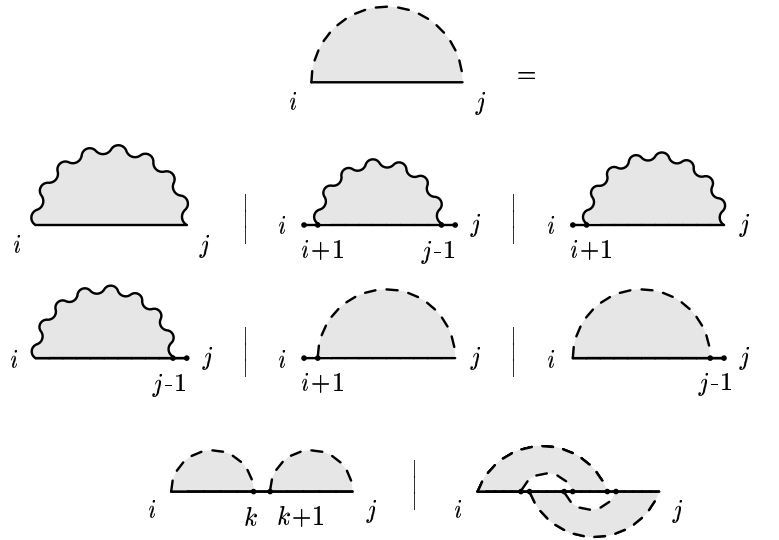


Figure 10: Recursion for wx in the pseudoknot algorithm truncated at $\mathcal{O}(whx + whx + whx)$.

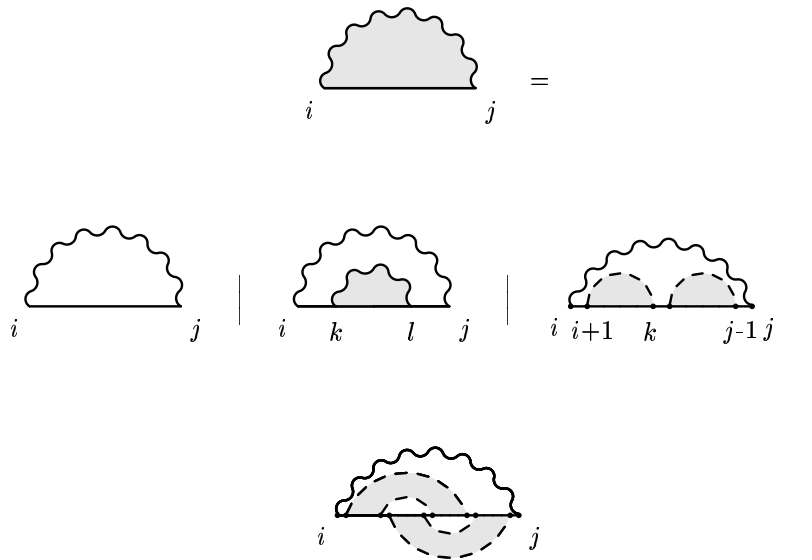


Figure 11: Recursion for vx in the pseudoknot algorithm truncated at $\mathcal{O}(whx + whx + whx)$.

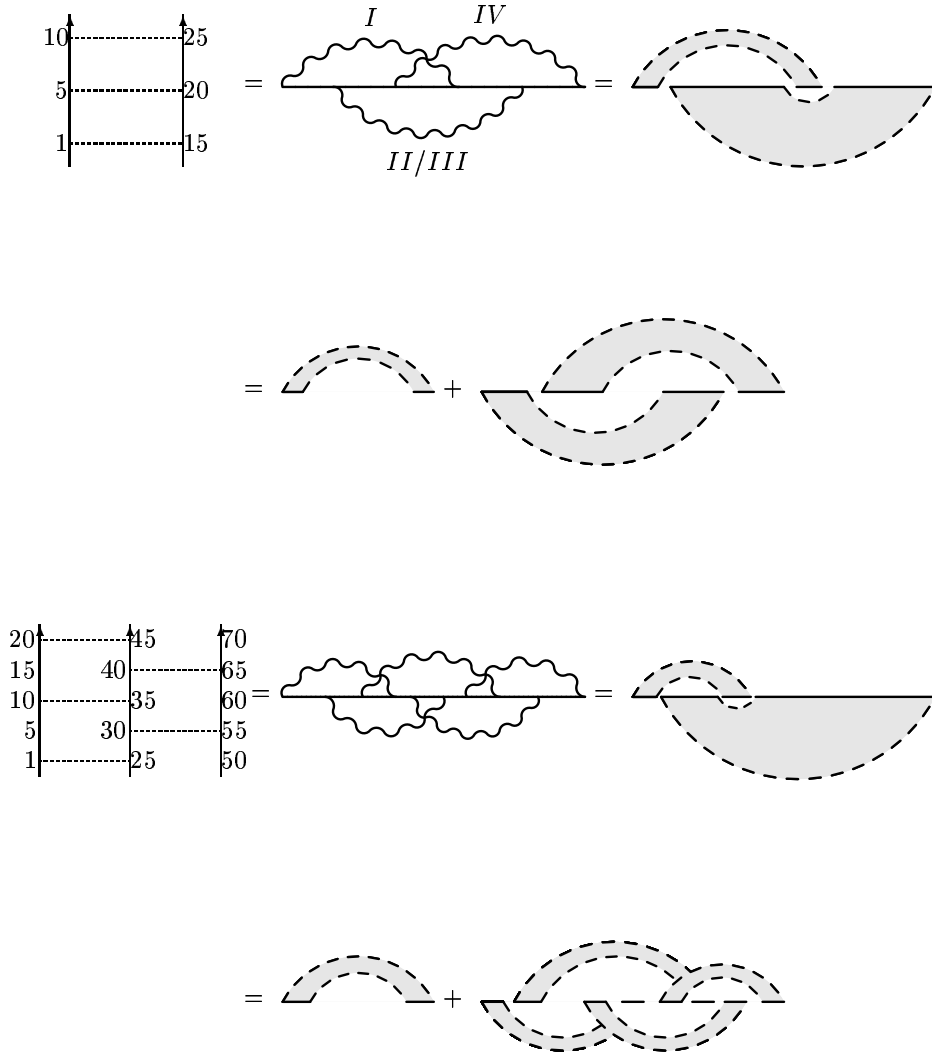


Figure 12: Top: The non-planar pseudoknot presented in α mRNA and how to build it with gap matrices. The Roman numbers correspond to the numbering of stems introduced by Gluick *et al.* (1994). Bottom: An example of a pseudoknot that the algorithm cannot handle: interlaced interactions as seen in proteins in parallel β -sheet. The assembly of this interaction using gap matrices would require us to use four gap matrices at once which is not allowed by the approximation at hand.

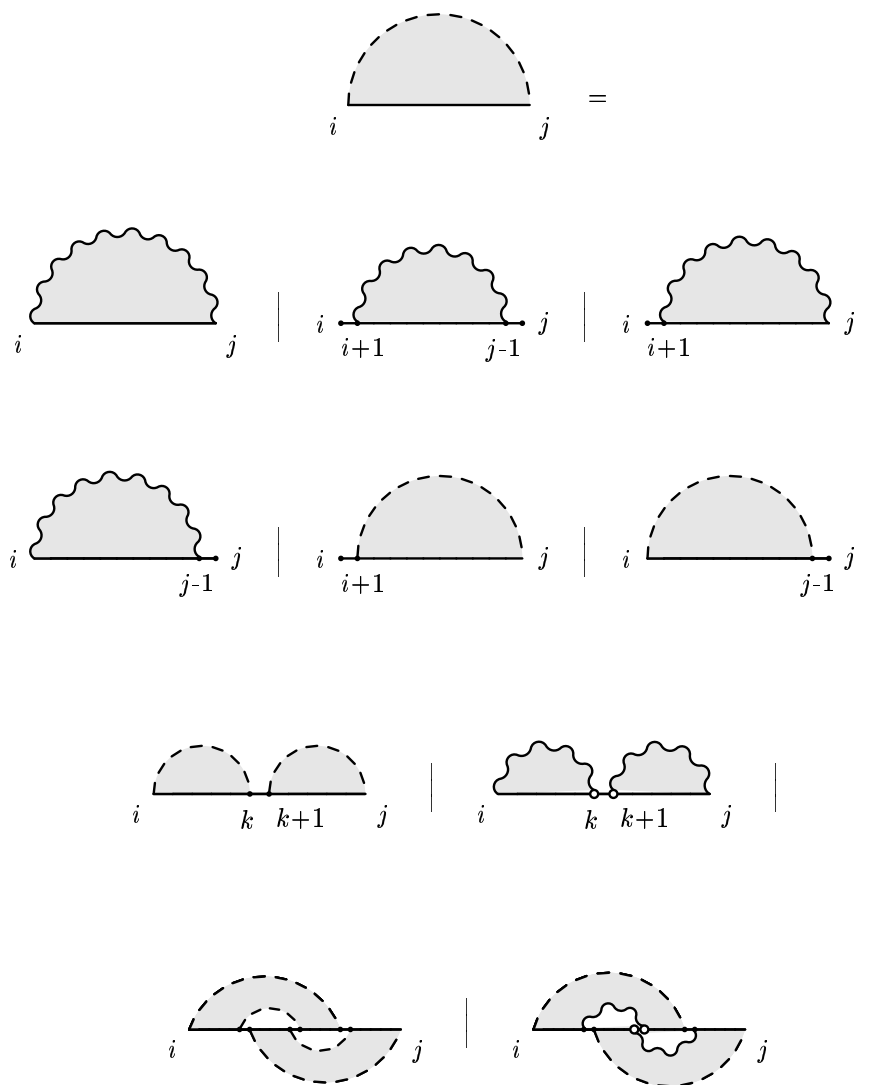


Figure 13: Recursion for the wx matrix.

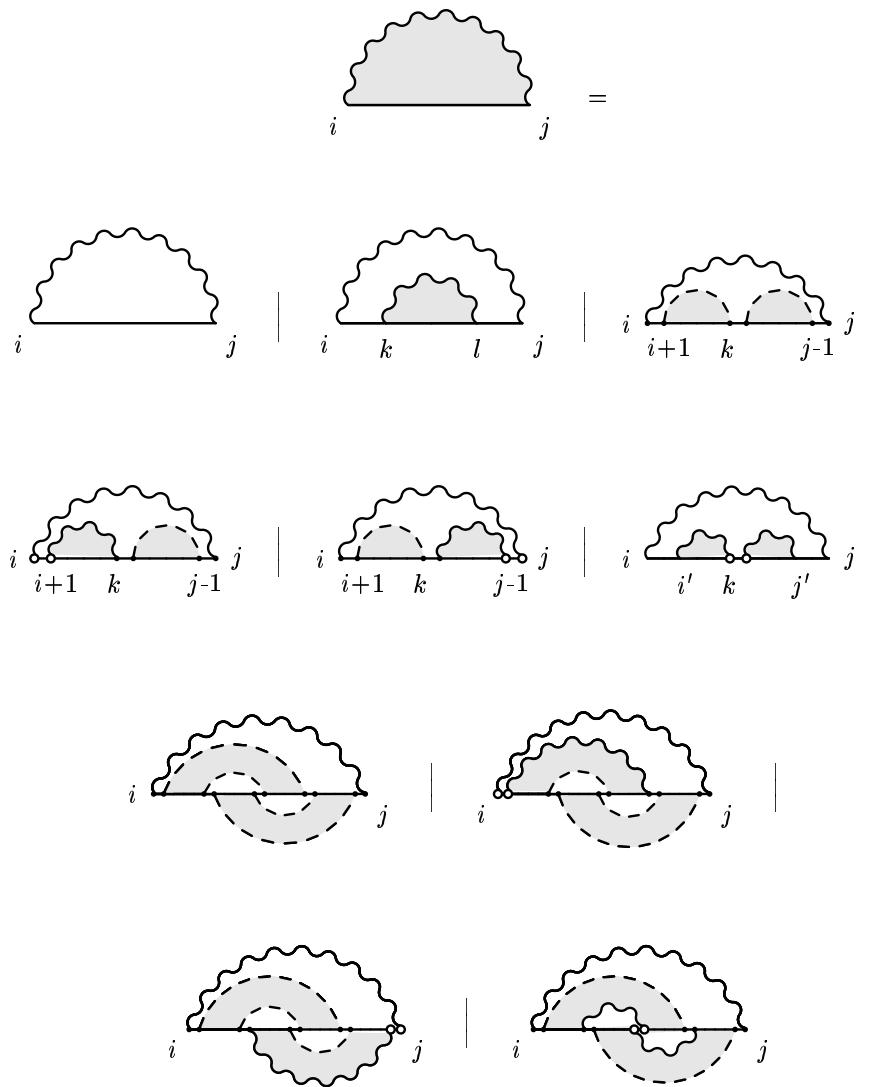


Figure 14: Recursion for the vx matrix.

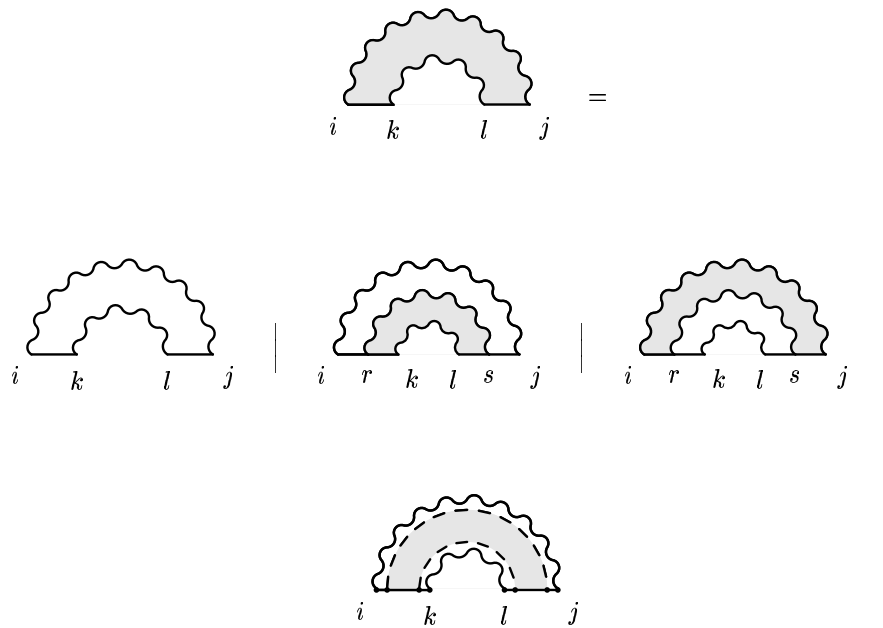


Figure 15: Recursion for the vhx matrix.

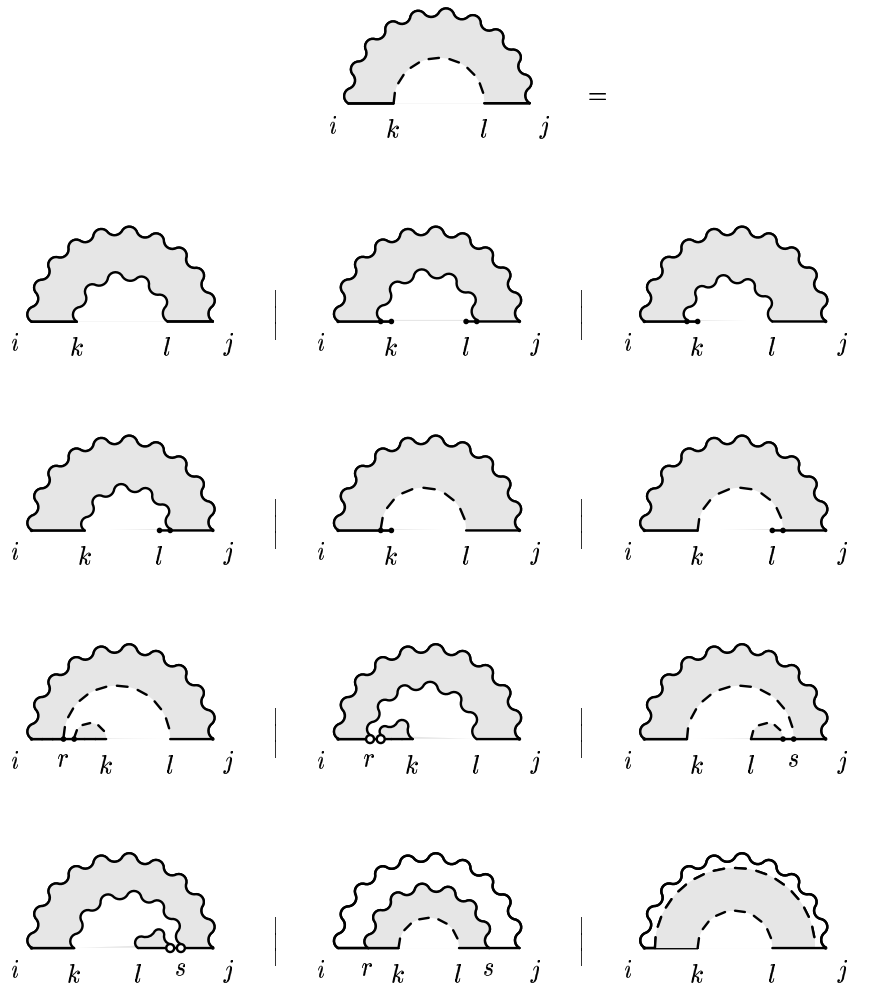


Figure 16: Recursion for the zhx matrix.

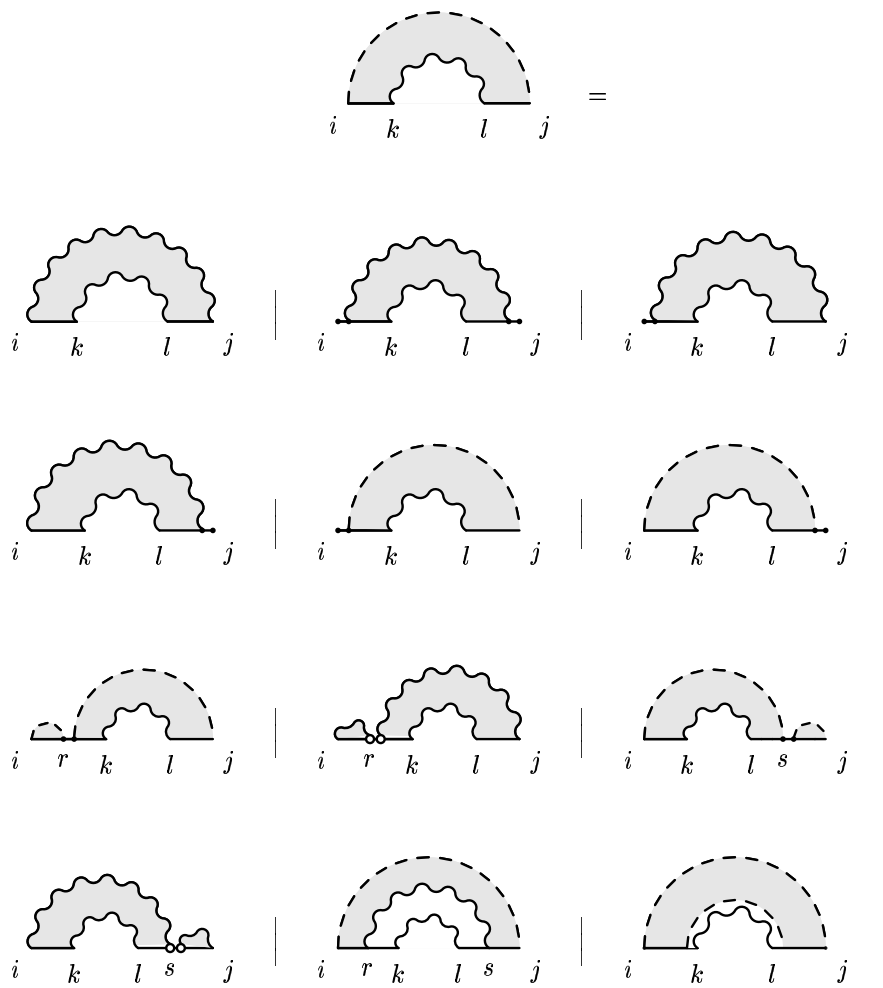
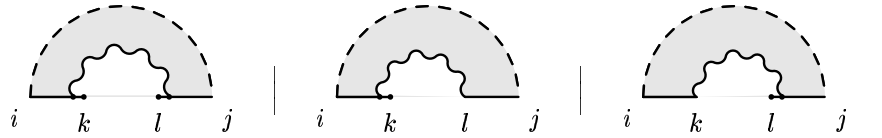
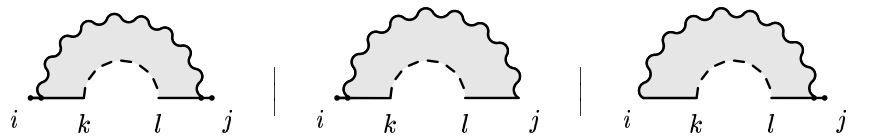
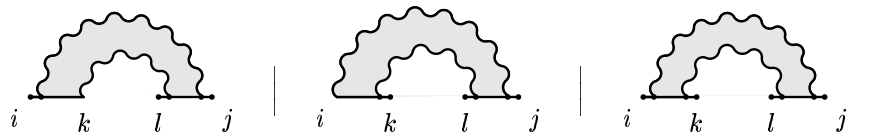
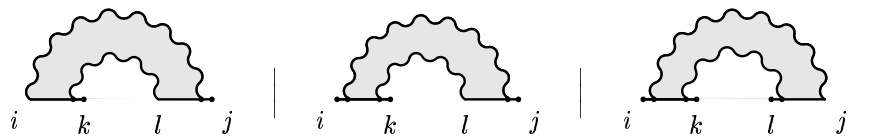
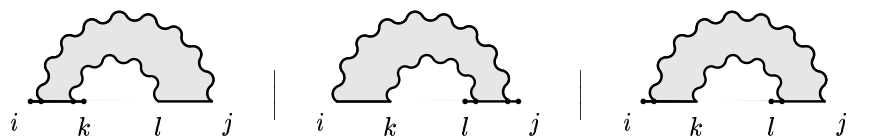
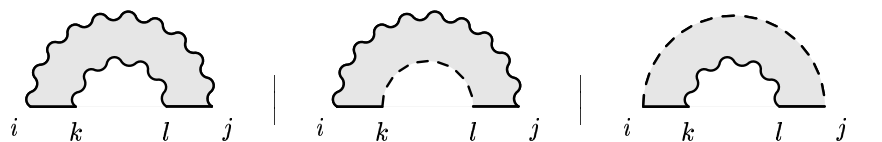
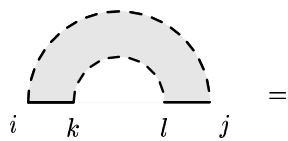
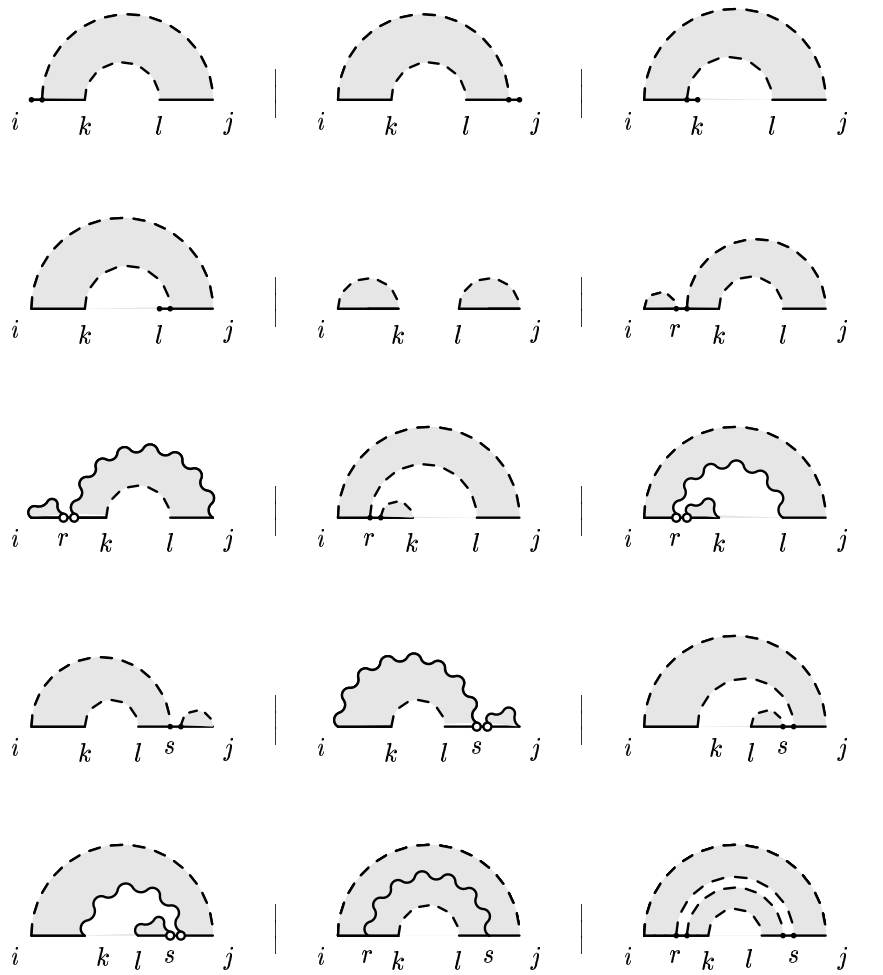


Figure 17: Recursion for the yhx matrix.





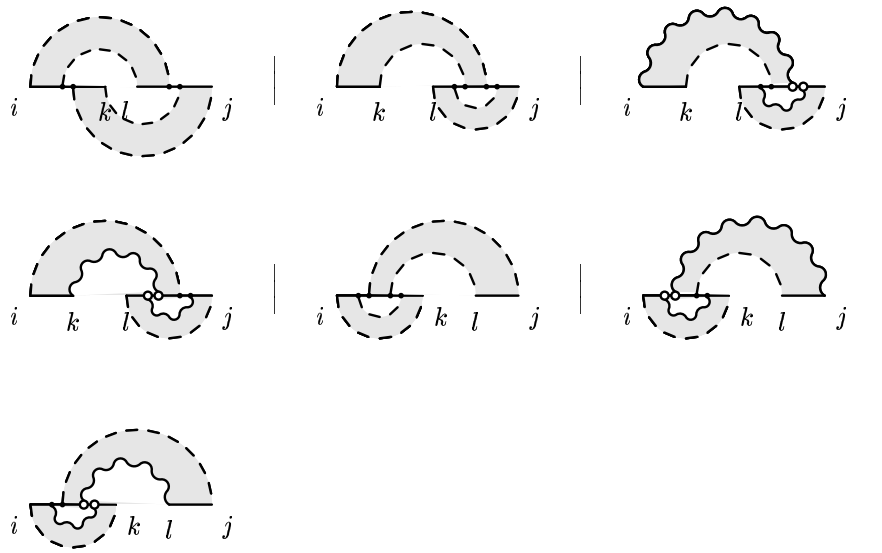


Figure 18: Recursion for the wx matrix.