Combinatorics of RNA secondary structures with base triples

Robert Müller[†] and Markus E. Nebel[†,‡]
{r_muelle, nebel}@cs.uni-kl.de
† Kaiserslautern University, Germany,
‡ Southern Denmark University, Denmark.

Abstract

The structure of RNA has been the subject of intense research over the last decades due to its importance for the correct functioning of RNA molecules in biological processes. Hence, a large number of models for RNA folding and corresponding algorithms for structure prediction have been developed. However, previous models often only consider base pairs, although every base is capable of up to three edge-to-edge interactions with other bases. Recently, Höner zu Siederdissen et al. presented an extended model of RNA secondary structure including base triples together with a folding algorithm – the first thermodynamics-based algorithm that allows the prediction of secondary structures with base triples. In this paper, we investigate the search space processed by this new algorithm, i.e. the combinatorics of extended RNA secondary structures with base triples. We present generalized definitions for structural motifs like hairpins, stems, bulges or interior loops occurring in structures with base triples. Furthermore, we prove precise asymptotic results for the number of different structures (size of search space) and expectations for various parameters associated with structural motifs (typical shape of folding). Our analysis shows that the asymptotic number of secondary structures of size n increases exponentially to $\Theta(\frac{4.10125^n}{n^{3/2}})$ compared to the classic model by Stein and Waterman for which $\Theta(\frac{2.61803^n}{n^{3/2}})$ structures exist. A comparison with the classic model reveals large deviations in the expected structural appearance, too. The inclusion of base triples constitutes a significant refinement of the combinatorial model of RNA secondary structure which by our findings is quantitatively characterized. Our results are of special theoretical interest, because a closer look at the numbers involved suggests that extended RNA secondary structures constitute a new combinatorial class not bijective with any other combinatorial objects studied so far.

1 Introduction and Basic Definitions

Ribonucleic acids (RNA) belong to the most abundant and important biological molecules involved in various functions ranging from catalyzing reactions to gene expression. An RNA molecule consists of a sequence of nucleotides connected via phosphodiester bonds. Every nucleotide comprises a nitrogenous base, a phosphate group and the pentose sugar ribose. There are four different bases: adenine (A), cytosine (C), guanine (G) and uracil (U), whereby A and U (respectively, C and G) are complementary. The sequence of bases in an RNA molecule is specific for it and thus represents its ‘primary structure’.

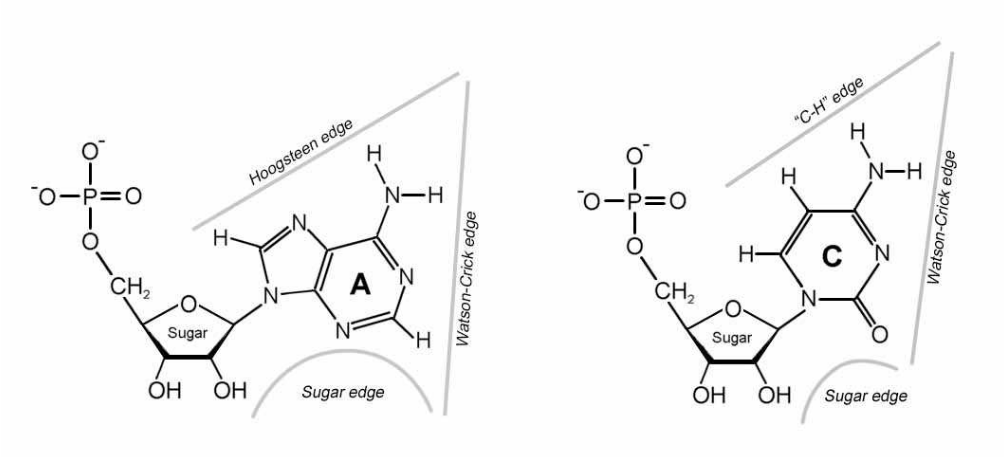


Figure 1: The three edges of possible base-to-base interactions of nucleotides containing a purine (left, adenine) or pyrimidine (right, cytosine). Based on Leontis and Westhof (2001, p. 501).

Non-neighboring nucleotides can interact via hydrogen bonds ('H bonds') between their bases. These base pairs are responsible for the folding of the RNA sequence into its three-dimensional conformation ('tertiary structure'), which is essential for its function. Traditionally, only complementary bases were known to pair up. Later, weaker forms of base pairing (so-called wobble pairs) have been discovered and today, 12 families of base pairs are characterized (Leontis and Westhof, 2001). The diversity of interactions stems from the possibility to form H bonds along three different edges (see Figure 1). Thus, a single base can interact with up to three other bases simultaneously. It has been discovered that about 40 % of all bases in structured RNA take part in edge-to-edge interactions other than canonical Watson-Crick base pairs (Leontis and Westhof, 2001). Base triples are a commonly found non-canonical interaction. They occur in RNA motifs such as sarcin-ricin loops and support tertiary RNA interactions (Abu Almakarem et al., 2012). Base triples are even part of a highly conserved, thus universal packaging mode of RNA helices (Doherty et al., 2001). Accordingly, the introduction of base triples into existing RNA models has the potential to bring them closer to nature. Especially the computational prediction of RNA structure which – for efficiency reasons – often considers a simplified, planar version of the tertiary conformation called secondary structure (we assume the reader familiar to the concept of secondary structures and refer to (Waterman, 1978; Hofacker et al., 1998; Nebel, 2002a) for details) can be expected to benefit from introducing base triples to the model. Figure 2 shows two examples of base triples occurring in real RNAs and illustrates that they can occur locally as well as between non-adjacent areas of the secondary structure. The arc diagrams shown in parts (a) and (c) of the figure assume the nucleotides to be represented by circles on a horizontal line (according to their ordering given by the sequence) while an arc in the upper half plane represents a hydrogen bond between the two incident nucleotides.

Predicting the secondary structure has been an ongoing topic of research for several decades resulting in a variety of models and algorithms (see e.g. Knudsen and Hein, 1999; Nussinov et al., 1978; Pipas and McMahon, 1975; Waterman, 1978; Zuker and Stiegler, 1981; Hofacker, 2003). However, they often do not consider base triples, but more recent research has started incorporating them. Höner zu Siederdissen et al. (2011) presented an extended model and introduced an accompanying dynamic programming version of the folding algorithm `MC-Fold` (Parisien and Major, 2008), called `RNAwolf`. Like the original version of

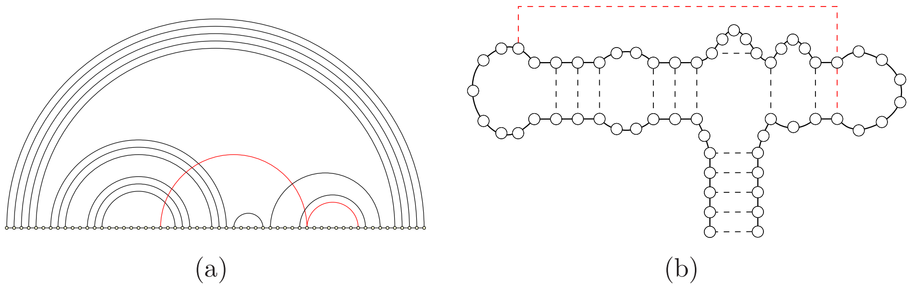
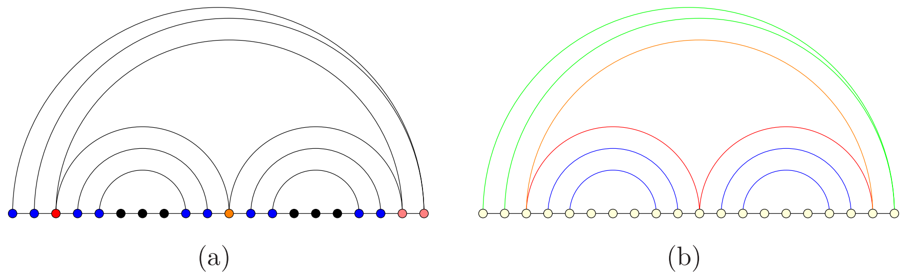


Figure 2: **(a)** Fragment of the 23S rRNA of *E. coli* showing a base triple connecting a stem and a non-adjacent hairpin loop and creating a pseudoknot (Conn et al., 1998). **(b)** Stem-loop representation of the 23S rRNA fragment shown in (a). **(c)** Fragment of the PDB structure `1dul`, which contains two combined base triples (red) as well as other non-canonical base pairs (blue) extending the secondary structure (Höner zu Siederdissen et al., 2011). **(d)** Stem-loop representation of PDB structure `1dul` shown in (c).

`MC-Fold`, `RNAwolf` considers non-canonical base pairings. In contrast to `MC-Fold` it is not able to handle pseudoknots but allows triple interactions, reducing the runtime from exponential to polynomial $\mathcal{O}(n^3)$. This elevates the model of extended RNA secondary structures from theoretically interesting to practically feasible through its applicability to RNAs of biological interest. However, only little is known about the changes implied by the incorporation of base triples to the combinatorics of RNA secondary structures. How many different extended structures for size n exist? What is their typical appearance, i.e. how do structural features like hairpin loops, bulges or interior loops behave in expectation? How many nucleotides do form a single, resp. two hydrogen bonds? Similar questions have been intensively studied for RNA secondary structures and it is the aim of this paper to start a similar line of research for extended structures providing a deeper understanding of their structural behavior.
Therefore, in this paper, we take up the model by Höner zu Siederdissen et al. (2011) and provide a large number of precise asymptotics related to the number of extended RNA secondary structures and the quantitative behavior of their various structural motifs. In accordance with the model, bases are allowed to link up with two other bases and we do not allow pseudoknots to occur. However, we do not distinguish between the different types of base-to-base interactions. Furthermore, as usual, we abstract from the actual primary structure of the RNA molecule assuming each possible folding to be equally likely. This is achieved by introducing an appropriate generalization of the well-known *dot-bracket* representation of secondary structures (see (Hofacker et al., 1998)).

1.1 Definitions

For alphabet Σ, we will write $\Sigma^\star$ (resp. Σ^+) to denote the set of all strings constructed from symbols in Σ including (resp. excluding) the empty word ε. The length of a string $w \in \Sigma^\star$



(a) (b)

Figure 3: (a) The different base types in our model: unpaired bases (black, symbol $*$), bases with exactly one H bond (blue, symbols $($, $)$) and bases with two H bonds connected to two bases downstream (red, symbol $(\!\!($), upstream (pink, symbol $)\!\!)$) or one in each direction (orange, symbol $)\!\!($). **(b)** H bonds can participate in simple pairs (blue) or in base triples. The H bonds colored in green form a simple base triple, while the red-colored ones highlight a so-called 'bridge'. A combination of red and orange H bonds denote a motif called 'cycle'.

is denoted by $|w|$, we write $|w|_a$ to count the number of occurrences of $a \in \Sigma$ in w. For a non-negative integer n, Σ^n represents all strings in $\Sigma^\star$ of length n. Analogously, for $a \in \Sigma$ we write a^n for the string build of n symbols a. Given $w \in \Sigma^\star$ we represent the ith symbol of w by w_i. We assume the reader to be familiar with the concept of context-free grammars (see (Harrison, 1978; Hopcroft et al., 1979) for details). We will interchangeably use the terms nonterminal and intermediate symbol to address a symbol that can be replaced by a production (or rule) of a grammar. We make use of the convention to use capital letters as nonterminals. A production f will be written in the form $f : A \to \alpha$ (omitting name f if not needed), representing the possibility to replace A by α. We will call A the premise or left-hand side and α the conclusion or right-hand side of f.

In order to represent nucleotides with more than one hydrogen bond we make use of the following alphabet

$$\Sigma := \{*, (,), (\!\!(,)\!\!),)\!\!(\},$$

whose symbols will also be called nucleotides or bases for their use to represent RNA molecules. The first three symbols in Σ are used in the same way as for classic dot-bracket words for RNA secondary structures; a dot $*$ represents an unpaired nucleotide while an opening (resp. closing) bracket stands for a nucleotide being bond to another one more to the right (resp. left) within the backbone. Accordingly, $(\!\!($ – consisting of two overlayed opening brackets – represents a nucleotide having two hydrogen bonds each with a nucleotide located to the right. In the same way $)\!\!)$ (resp. $)\!\!($) represents a nucleotide with two hydrogen bonds to the left (resp. with one to the left and one to the right). Figure 3 (a) shows the corresponding base types using arc diagrams.
Having this interpretation in mind, we are now ready to provide a formal definition of extended RNA secondary structures. As in (Kemp, 1996) we make use of a congruence relation to define the structural properties shared by all words of the language to be defined – a generalized Dyck-language in case of (Kemp, 1996) with obvious ties to dot-bracket words.

Definition 1 (Extended secondary structures) *Let* $\Sigma = \{*, (,), (\!\!(,)\!\!),)\!\!(\}$ *and denote by* δ *the congruence defined by the following set of equations:* $\{* \equiv_\delta \varepsilon,\ (\,) \equiv_\delta \varepsilon,\ (\!\!(\,) \equiv_\delta (,\ (\!\!(\,)\!\!(\equiv_\delta ((,\ (\,)\!\!) \equiv_\delta),\ )\!\!(\,)\!\!) \equiv_\delta)),\ )\!\!(\,)\!\!(\equiv_\delta)(,\ ()\!\!(\equiv_\delta (,\ )\!\!() \equiv_\delta)\}$ *for* ε *the empty string. Then the set of* extended RNA secondary structures *is defined by language* $\mathcal{L} \subseteq \Sigma^\star$ *with* $w \in \mathcal{L}$ *if and only if* w *is equivalent to* ε *under congruence* δ *and furthermore* w *is only congruent to itself if we are not allowed to apply equation* $* \equiv_\delta \varepsilon$*, i.e.,* $w \equiv_{\delta \setminus \{* \equiv_\delta \varepsilon\}} \rho \implies \rho = w$.

We can think of equation $* \equiv_\delta \varepsilon$ to allow the deletion of all the unpaired nucleotides (symbols *) from our representation. Afterwards, using any of the other congruence equations corresponds to the deletion of an arc within the associated arc diagram. Here symbols are changed, not deleted, if base triples are involved, reflecting the idea of a step-by-step deletion of any bond. For example, applying equation ⋈⋈ $\equiv_\delta$)(is assumed to delete the bond between two nucleotides with base triples, leaving them both with a single bond. In the sequel, we will call the respective remaining symbol (i.e., those one right-hand side of the equation applied) the *residue* of the original one. For example, in connection with the application of ⦅⋈ $\equiv_\delta$ ((, the first ((resp. second)) is the residue of ⦅ (resp. ⋈). Accordingly, we call two bracket symbols of an extended secondary structure w *corresponding with respect to* w, if a congruence equation is applied to themselves or to their residues of a former application in order to derive $w \equiv_\delta \varepsilon$ (i.e. if the equivalent arc diagram has an arc connecting the two related vertices). While deriving $w \equiv_\delta \varepsilon$, in order to be allowed to apply a congruence equation it is required that the bracket symbols on the left-hand side of the equation are neighbored. This is equivalent to the assumption that an arc cannot be deleted before all arcs (and nucleotides *) below it have been erased. As a consequence, arc diagrams equivalent to w with $w \equiv_\delta \varepsilon$ cannot have crossing arcs; this way our definition excludes pseudoknots. Demanding w to be only congruent to itself in case * cannot be deleted implies a minimal (hairpin) loop length of one; a * in between any hairpin closing pair of (generalized) brackets prevents any arc from being erased (for $\equiv_{\delta\setminus\{* \equiv_\delta \varepsilon\}}$). Note that ⦅⦆ does not contribute an equation to the congruence for we do not allow two nucleotides to have two hydrogen bonds with each other.

Example 1 *The strings* ⦅()*)) *and* (*⋈*⋈*⋈)) *represent no extended secondary structure for the first is congruent* ⦅*)) *even if equation* $* \equiv \varepsilon$ *is not applied and the second is congruent*) *but not congruent* ε*. However,* $w =$ ((*⋈((*)⦆)** *represents a valid extended secondary structure which by the steps* ((*⋈((*)⦆)** $\equiv_\delta$ ((⋈(()⦆) $\equiv_\delta$ ((⋈(⦆) $\equiv_\delta$ (((⦆) $\equiv_\delta$ (()) $\equiv_\delta$ () $\equiv_\delta \varepsilon$ *can be reduced to* ε*. Thus, the first* (*and the last*) *of* w *are corresponding with respect to* w *since we applied* () $\equiv_\delta \varepsilon$ *in the last step. As a second example consider the arc diagram of Figure 6 which corresponds to the extended secondary structure* (⦅⦅***)⦆⦆.

In order to prevent confusion, we will call a secondary structure without base triples *classic structure* in the sequel. We then assume standard dot-bracket words to be used for their representation.

As we will see in the next section, allowing base triples implies a huge growth in the number of different structures of given size. But not only the number of structures is changed, also the set of possible structural features is enriched and motives like stems and hairpin loops need to be redefined. However, it will be possible to formally reduce the new motifs to the corresponding traditional ones (for background information on different structural motifs in the classic model without base triples, we refer the reader to Sankoff and Kruskal (1999, Chapter 3)), thus putting the definition of the new motifs down to that for classic structures. To this end we will make use of the *expansion* of an extended secondary structure defined as follows:

Definition 2 *Let* $w \in \Sigma^\star$ *an extended secondary structure and let* e *denote the morphism defined by* $e($⦅$) =$ ((, $e($⦆$) =$)), $e($⋈$) =$)(, *and* $e(x) = x$ *for* $x \in \{$(,), *$\}$*. Then* $e(w)$ *is called the* expansion *of* w*.*

In order to make use of expansions to define structural motifs, we will need to determine the preimage of a substring of an expansion. Here we stick to the convention, that every symbol of an extended secondary structure w belongs to the preimage of a substring u of $e(w)$ that

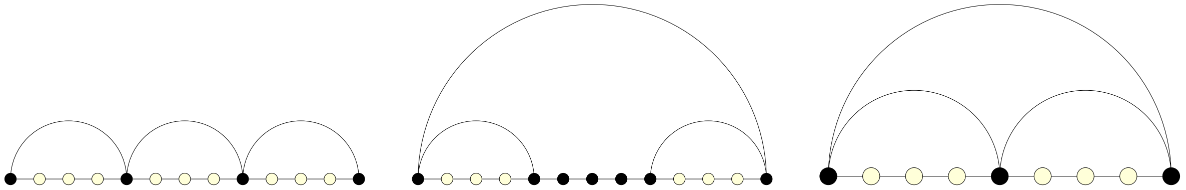


Figure 4: All types of novel hairpin loops, i.e. hairpin loops not nested directly inside a simple base pair. Every unbroken sequence of bright bullets represents a hairpin loop.

contributes at least one symbol to u under morphism e. For example, with $w =$ ⸨*)*) we have $e(w) =$ ((*)*) and accordingly ⸨*) is the preimage of substring (*).
Note that not every string w over $\Sigma = \{$*, (,), ⸨, ⸩, X$\}$ with $e(w)$ a classic structure represents an extended secondary structure. The reason is simple: for extended structures we do not allow nucleotides ⸨ and ⸩ to represent two hydrogen bonds joining both the same nucleotides – thus no equation ⸨⸩ $\equiv \varepsilon$ – but based on the expansion we cannot prevent ⸨⸩ to be assumed corresponding.
Now we are ready to define generalized structural motifs: according to (Nebel, 2004), a classic structure $w = w_1 \cdots w_n$, $w_i \in \{$(),*$\}$, $1 \leq i \leq n$, has a hairpin loop chl v iff $v = w_{i+1} \dots w_{j-1}$ with $v = *^{j-i-1}$ and $w_i w_j =$ () hold. We will identify hairpin loops for an extended structure with those chl found in its expansion.

Definition 3 (Hairpin loops) *Let $w \in \Sigma^\star$ an extended secondary structure. A substring $v = w_{i+1} \dots w_{j-1}$ of w with $v \in \{*\}^+$ is called* hairpin loop *(of length $|v|$) iff there exists substring u of $e(w)$ which is a* chl *(for $e(w)$) and v is the preimage of u. If $e(w_i v w_j) = w_i u w_j$ we call the hairpin loop* classic*, otherwise* novel*.*

Thus, a hairpin loop is classic if any only if it is enclosed in a pair of brackets () (lies inside a simple base pair). If base triples are involved, new kinds of hairpins result. Figure 4 shows the resulting different types of novel hairpin loops.
A classic structure w has a left (resp. right) bulge cb iff it contains a non-empty subword $v = w_{i+1} \dots w_{j-1}$ with $v = *^{j-i-1}$ and $w_i w_j =$ (((resp. $w_i w_j =$))) and there exists k with $w_k w_{k+1} =$)) (resp. $w_k w_{k+1} =$ (() such that w_j, w_k and w_i, w_{k+1} are corresponding. A similar motif is given by interior loops which essentially consist of a left and a right bulge which are located face to face with each other. Precisely, two non-empty subwords $v_1 = w_{i+1} \dots w_{j-1}$, $v_2 = w_{k+1} \dots w_{l-1}$ of a classic structure w form an interior loop cil iff $v_1 = *^{j-i-1}$, $v_2 = *^{l-k-1}$ and $w_i w_j =$ ((, $w_k w_l =$)) such that w_i, w_l and w_j, w_k are corresponding.

Definition 4 (Bulges and Interior loops) *Let $w \in \Sigma^\star$ an extended secondary structure. Two substring v_1, v_2 of w form an* interior loop *(of length $|v_1| + |v_2|$) iff there exist substrings u_1, u_2 of $e(w)$, u_1, u_2 a* cil *(for $e(w)$), where v_i is the preimage of u_i, $i \in \{1, 2\}$. A substring v of w is a* bulge *(of length $|v|$) iff there exists a substring u of $e(w)$ being a* cb *(for $e(w)$) for which v is the preimage.*

According to this definition, bulges and interior loops (Figure 5 shows an example for both) are counted independently, i.e. an interior loop does not contribute two bulges. Note, that hairpin loops, bulges and interior loops are not sufficient to cover all occurrences of unpaired nucleotides. For example, *tails*, i.e., runs of symbol * at the beginning or ending of a secondary structure do not fall into any of those categories.
Last but not least, a classic stem cs consists of two non-empty maximal subwords y, z such that $y = w_i \dots w_{i+c}$ and $z = w_j \dots w_{j+c}$ and $w_{i+k} w_{j+c-k}$ is a pair of corresponding brackets, $0 \leq k \leq c$ (see (Nebel, 2004)).

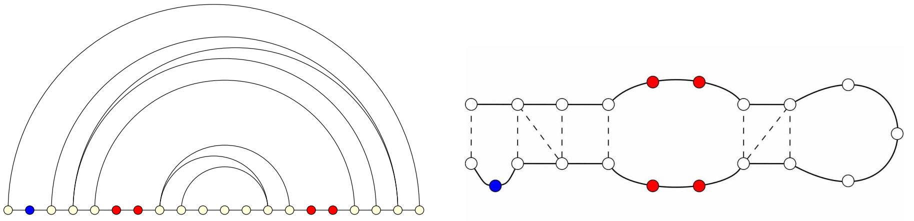


Figure 5: **(a)** Extended RNA secondary structure containing both a bulge (blue) and an interior loop (red). **(b)** Stem-loop representation of the extended secondary structure in (a).

Definition 5 (Stems) *Let $w \in \Sigma^\star$ an extended secondary structure. Two substrings v_1, v_2 of w form a* stem *iff there exist a* cs *$u_1 =$ `(`$^{c+1}$, $u_2 =$ `)`$^{c+1}$ in $e(w)$ such that the preimage of u_1 (resp. u_2) is v_1 (resp. v_2). If $v_i = u_i$, $i \in \{1, 2\}$, the stem is called* classic, *otherwise* novel.

Note that a symbol $b \in \{$⦅, ⦆, X$\}$ can contribute to two stems in cases where the two brackets of $e(b)$ are part of two different classic stems (for $e(w)$). In the example $w =$ `⦅*)*)` from above, we have $e(w) =$ `((*)*)` and ⦅ is part of the preimage of two cs for $e(w)$ (the inner and the out corresponding pair for brackets); the corresponding nucleotide is part of two stems, contributing one hydrogen bond to each.

Example 2 *Consider $w =$ `(⦅*)*⦆`. This extended secondary structure contains two stems. We find for the expansion $e(w) = \underline{((}\overline{(}*\overline{)}*\underline{))}$ containing two stems highlighted by under- resp. overlined symbols. The respective preimages v_1 and v_2 for the underlined stem are given by $v_1 =$ `(⦅` and $v_2 =$ `⦆`, that for the overlined one by $v_1 =$ `⦅` and $v_2 =$ `)`. Thus symbol ⦅ contributes to both stems.*

Before definition nicely relates stems in extended secondary structures to those in classic structures. However, it sheds no light on the appearance of novel stems. To this end, we need to introduce a novel structural motif called *bond chain*. For its definition we will not make use of the concept of expansions for not to hide the true structural appearance.

Definition 6 (Bond chains) *Let $w \in \Sigma^\star$ an extended secondary structure and let v a substring of w with $v \equiv_\delta \varepsilon$ such that there exist $b_i \in \{$`(`, `)`, ⦅, ⦆, X$\}$, $1 \leq i \leq k$, $k \geq 2$, and $u_j \in \Sigma^\star$, $1 \leq j < k$, with $v = b_1 u_1 b_2 u_2 \cdots u_{k-1} b_k$. We call the subsequence[1] of w identified with $b := b_1 b_2 \cdots b_k$ a* (hydrogen) bond chain, *iff $u_j \equiv_\delta \varepsilon$, $1 \leq j < k$, and $b \equiv_\delta \varepsilon$ applying congruence `()` $\equiv_\delta \varepsilon$ exactly once. Any subsequence of b consisting only of symbols that are corresponding w.r.t. w is considered a* partial *bond chain.*

In the sequel we will speak of a *segment* to address an unpaired base or a bond chain. Accordingly, any extended secondary structure can be decomposed into its segments. Please note that the assumption $v \equiv_\delta \varepsilon$ of before definition implies the maximality of any bond chain b in the sense that we cannot extend the considered substring to the left or right in order to obtain a bond chain b' for which b is a substring. With respect to arc diagrams, bond chains correspond to the non-singleton connected components (disregarding the backbone with respect to adjacency). The length of a bond chain is given by the number of arcs in the corresponding connected component. According to before definition, the simplest bond

[1] Note the usage of a subsequence which allows symbols of w to be left out.

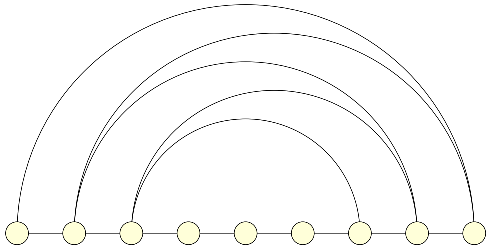


Figure 6: Exemplary H bond chain with a typical, but not necessary zigzag pattern.

chain consists of a single pair, i.e., in terms of arc diagrams, of two vertices connected by an arc. In general, it is possible to draw all the arcs of a bond chain "without lifting the pen" (thus its name). We call a bond chain *zigzag* if our drawing direction (left/right) changes during that task; an example is shown in Figure 6. For partial bond chains only those arcs are considered with respect to zigzags whose open and closing bracket both are part of the corresponding substring. There are two special kinds of bond chains we need to distinguish in connection with stems of extended secondary structures:

Definition 7 (Bridges and Cycles) *Let $w \in \Sigma^\star$ an extended secondary structure and let b a potentially partial bond chain of w. We call a (not necessarily proper) subsequence v of b a* bridge *(of length l) iff $v \in \{(, ⸨\} \cdot \{X\}^{l-1} \cdot \{), ⸩\}$, $l \geq 2$, $v \equiv_\delta \rho$, $\rho \in \{\varepsilon, (,), ()\}$, applying δ only to symbols that are corresponding w.r.t. w and no residue of symbol X shows up in ρ. We call b* cycle *(of length $l+1$) iff $b =$ ⸨ $\cdot \{X\}^{l-1} \cdot$ ⸩, $l \geq 2$, and $b \equiv_\delta \varepsilon$, appyling δ only to symbols that are corresponding w.r.t. w (which prevents b from being partial).*

Note that a cycle is a special kind of a bridge and thus contributes to both motifs for counting. Examples for bridges are given by the subsequences (XX) (with $\rho = \varepsilon$) or ⸨XX) (with $\rho =$ (, being a substring of the bond chain ⸨XX))), an example for a cycle is ⸨XXX⸩.

With those terms we are able to express the appearance of novel stems in a descriptive manner: A stem consists of directly nested bond chains which are not interrupted by unpaired bases. If a bond chain is interrupted by an unpaired nucleotide so are the runs of opening or closing brackets in its expansion thus giving rise to several stems. Furthermore, bridges (and thus also cycles) containing symbol X are only allowed to show up at the inner or outermost position of a stem. The reason is obvious: symbol X corresponds to sort of a branching of a stem into two which should be considered the starting point of new stems. Accordingly, a nucleotide represented by X within a limiting bridges is part of two stems and contributes a hydrogen bond to each of them. Figures 7 and 8 show novel stems containing bridges and cycles.

The rest of the paper is structured as follows. In Section 2, we present the results of our asymptotic analysis and reason about their importance. First, the number of extended RNA secondary structures is considered. Subsequently, we investigated the structural motifs just defined and present results for their averages, proportions and lengths. Section 3 offers more details on the used methodology. In Section 4, we conclude with summarizing our main findings and suggesting future extensions of our work.

2 Results

In this section, we present our results without describing the technical details of their proofs – those are presented in Section 3.

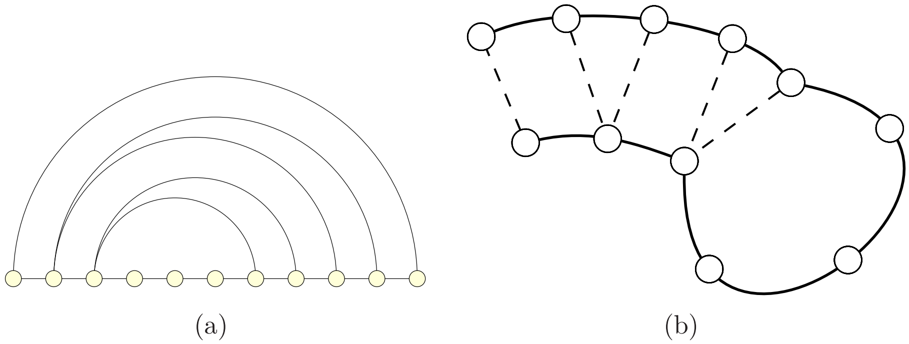


Figure 7: **(a)** One kind of novel stem consisting mainly of simple base triples causing a bent appearance of the stem. **(b)** Stem-loop representation of the bent stem in (a).

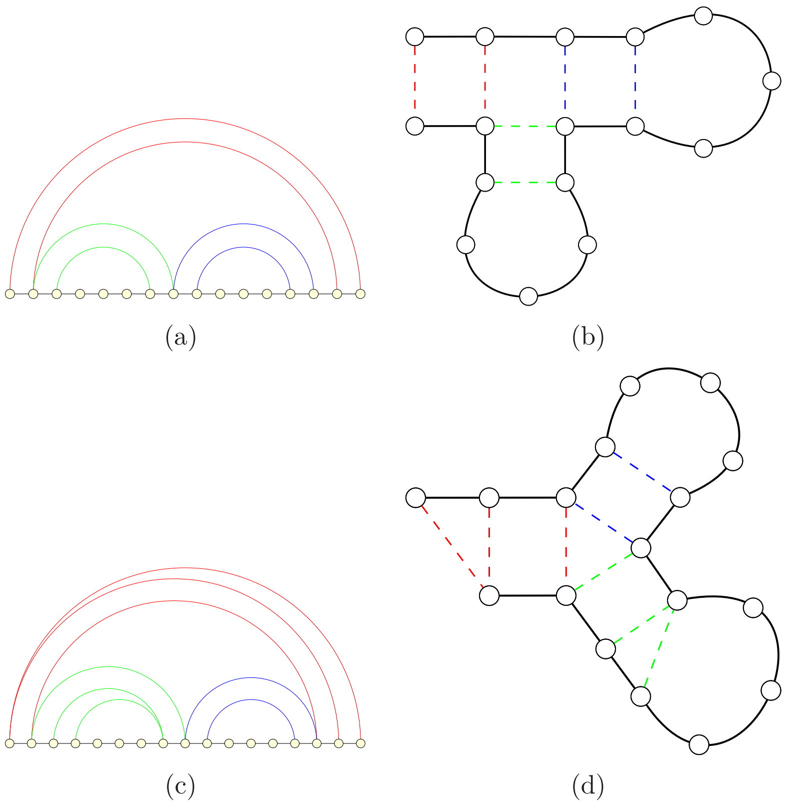


Figure 8: Novel stems starting from (a) bridge structures and (c) cycles and their corresponding stem-loop representations (b) resp. (d) where the hydrogen bonds of each stem are colored differently.

Table 1: Comparison of asymptotic and exact coefficients for different lengths of RNA sequences.

Length	Asymptotic coefficient	Exact coefficient	Quotient
10	10667	9402	1.13455
25	4215604519670	4007586822784	1.05191
50	$3.1348324558 \times 10^{27}$	$3.0562890692 \times 10^{57}$	1.02570
100	$4.9030720370 \times 10^{57}$	$4.8411551219 \times 10^{57}$	1.01279
250	$1.0738782841 \times 10^{149}$	$1.0684271986 \times 10^{149}$	1.00510
500	$6.4328693628 \times 10^{301}$	$6.4165155710 \times 10^{301}$	1.00255
750	$5.9328382569 \times 10^{454}$	$5.9227776464 \times 10^{454}$	1.00170
1000	$6.5290360859 \times 10^{607}$	$6.5207300849 \times 10^{607}$	1.00127

2.1 Counting extended RNA secondary structures

First of all, we investigated the number of different extended secondary structures of size n according to the model presented in (Höner zu Siederdissen et al., 2011). Initially we determined the exact numbers up to length 1000. The corresponding counting sequence starts as follows:

> 1, 1, 2, 6, 20, 66, 221, 757, 2647, 9402, 33813, 122876, 450543, 1664835, 6193553, 23178525, 87199503, 329592705, 1251029482, 4766563155, 18223733493, 69892469057, 268824143700, 1036690445777, 4007586822784, . . .

A lookup in the 'On-Line Encyclopedia of Integer Sequences'[2] did not result in a match. This suggests that pseudoknot-free RNA secondary structures with base triples are a new combinatorial object and even not bijective with any other combinatorial objects studied before[3].

Subsequently, we computed the precise asymptotic number of extended secondary structures:

$$\frac{0.25053615497618946335 \times 4.10124749999982084857^n}{\sqrt{n^3}} \tag{2.1}$$

Table 1 shows the result of a comparison between exact and asymptotic numbers for selected sizes up to 1000 bases. Even for small n like those appropriate for RNA families such as tRNAs (about 75 bases), the relative error is only between 1 and 2 %. Larger values of n convenient for, e.g., ribozymes such as RNase P face deviations of clearly less than 1 %.

Compared to the seminal result by Stein and Waterman (1979) one sees a massive gain in the number of secondary structures. The precise asymptotic number of classic structures of size n is given by $1.10437\, n^{-3/2} \times 2.61803^n$. This shows that the introduction of base triples leads to an exponential increase by a factor of 0.22686×1.56654^n.

Qin and Reidys (2007) examined another extension of RNA secondary structures including both base triples and pseudoknots. Using bijections and the theory of Birkhoff-Trjitzinksy, they obtained an asymptotic formula with exponential growth 8^n for the number of different conformations in their model. An example for a corresponding structure is shown in Figure 2(a). Compared to our approach their model includes only a subset of possible base

[2] See `http://oeis.org/`.

[3] Note that the same holds when considering extended secondary structures with a minimal hairpin loop length of 0, which might be assumed a more natural structure from a pure combinatorial point of view.

triples interactions: When reducing their model to not allowing pseudoknots, the number of remaining structures of size n corresponds to the nth Motzkin number[4] (Chen et al., 2008; Bousquet-Mélou and Xin, 2006). Thus, it is significantly reduced to a size with exponential rate of growth 3^n proving our model to provide a much richer (pseudoknot-free) extension of classic RNA secondary structures.

2.2 The expected quantitative behavior of structural motifs

We examined the structural motifs defined above in terms of their number of occurrences and, if applicable, their length. Here the length could either be measured in the number of bases or the number of hydrogen (H) bonds that contribute to a motif – sometimes both. Finally, we derived three types of expectations: averages per extended secondary structure, proportions and lengths.

Table 2 shows the results of counting the motifs and determining their averages per extended secondary structure. As can be seen from this table, all means grow linearly with the length of the RNA sequence. However, there is a rather wide range of coefficients. They vary from 0.00564 for the number of interior loops up to 0.64496 for the number of bases in all stems.

In Table 3 proportions of various kinds for the different parameters are presented. It is noticeable that there is a relatively even distribution of bases with zero, one and two H bonds. In contrast, the subtypes show much more variation in their distributions. Take, for example, the case of bases with a single H bond. These are much more often connected to a base triple than to another simple base (70 % and 30 %, respectively). Stems dominate the RNA sequences in our model and account for nearly two out of three bases. Hairpin loops on the other hand make up only 20 % of all bases. However, they represent more than half of all unpaired bases. The main contributor to the number of H bonds are those occurring in base triples with a proportion of almost 90 %. While cycles only account for 5 % of all H bonds, bridge structures contain almost 30 % of all H bonds.

Table 4 displays expected lengths of different examined motifs. A comparison between the minimum and expected lengths reveals that the motifs on average barely exceed the minimum. Unsurprisingly, all motifs consisting of a sequence of unpaired bases share their expected length, because they – as will become obvious during our proofs – are all based on the same structural decomposition. The only exception is the expected length of interior loops, which results from the existence of two unpaired sequence per interior loop. Contrary to our expectations, the novel stems are only slightly longer than the classic stems (2.67691 bases vs. 2.12642 bases). Although novel stems result from base triples, they do not have a minimum length of two. Some novel stems, e.g. one arc inside a bridge structure, comprise only a single H bond.

A comparison between our model and corresponding results for classic structures reveals some notable differences (Table 5). In our model, there are on average less unpaired bases per secondary structure and we have less bulges and more hairpins, while both decrease similarly in length. In contrast, we have on average more H bonds in general as well as more

[4] The nth Motzkin number m_n is given by the number of strings $w \in \{*, (,)\}^n$ with $|w|_(= |w|_)$ and for which any factorization $w = u \cdot v$ implies $|u|_(\geq |u|_)$ (Nebel, 2004); we have $m_n \sim \frac{3\sqrt{3}}{2\sqrt{\pi n^3}} 3^n$, $n \to \infty$.

Table 2: Precise asymptotics for the average value per extended secondary structure for various parameters together with the native behavior derived from PDB data. Variable n represents the size of the extended secondary structures considered.

Parameter	**Expectation (combinatorics)**	**Expectation (database)**
Number of unpaired bases	$0.35504n$	$0.159068n$
Number of bases (1 H bond, total)	$0.33020n$	$0.834236n$
Number of bases (1 H bond, within simple pairs)	$0.09852n$	$0.824703n$
Number of bases (1 H bond, outside simple pairs)	$0.23168n$	$0.009534n$
Number of bases (2 H bonds, total)	$0.31476n$	$0.006696n$
Number of bases (2 H bonds, center position)	$0.07509n$	$0.000857n$
Number of bases within bridges	$0.20444n$	$0.002551n$
Number of runs of unpaired bases	$0.26847n$	$0.090800n$
Number of cycles	$0.00794n$	$-0.000011n$
Number of bridges	$0.06467n$	$0.000850n$
Number of bond chains	$0.17304n$	$0.413517n$
Number of segments	$0.52808n$	$0.572585n$
Number of H bonds in cycles	$0.02509n$	$-0.000033n$
Number of H bonds in bridges	$0.13977n$	$0.001700n$
Number of H bonds in triples	$0.43060n$	$0.011463n$
Number of H bonds in simple pairs	$0.04926n$	$0.412351n$
Number of H bonds	$0.47986n$	$0.414673n$
Number of classic hairpin loops	$0.02791n$	$0.002802n$
Number of novel hairpin loops	$0.11916n$	$0.000318n$
Number of all hairpin loops	$0.14707n$	$0.003120n$
Number of bases in classic hairpin loops	$0.03692n$	$-0.000950n$
Number of bases in novel hairpin loops	$0.15758n$	$0.000306n$
Number of bases in all hairpin loops	$0.19450n$	$-0.000644n$
Number of classic stems	$0.02957n$	$0.067407n$
Number of novel stems	$0.29382n$	$0.004948n$
Number of all stems	$0.32339n$	$0.072355n$
Number of bases in classic stems	$0.06288n$	$0.768499n$
Number of H bonds in classic stems	$0.03144n$	$0.384250n$
Number of bases in novel stems	$0.58208n$	$0.072283n$
Number of H bonds in novel stems	$0.44842n$	$0.039565n$
Number of bases in all stems	$0.64496n$	$0.840782n$
Number of H bonds in all stems	$0.47986n$	$0.423814n$
Number of bulges	$0.07626n$	$0.021935n$
Number of interior loops	$0.00564n$	$0.028501n$
Number of bases in bulges	$0.10085n$	$0.038567n$
Number of bases in interior loops	$0.01491n$	$0.094708n$

Table 3: Expected proportions for different parameters.

Parameter	**Expectation**
Unpaired bases of all bases	0.35504
Simple base pairs of all bases	0.33020
Simple base pairs inside simple pairs of all simple base pairs	0.29837
Simple base pairs outside simple pairs of all simple base pairs	0.70163
Base triples of all bases	0.31476
Centered base triples of all base triples	0.23858
Bases within bridges of all bases	0.20444
H bonds in simple pairs of all H bonds	0.10266
H bonds in triples of all H bonds	0.89734
H bonds in cycles of all H bonds	0.05228
H bonds in bridges of all H bonds	0.29127
Bases in classic hairpin loops of all bases	0.03692
Bases in classic hairpin loops of all unpaired bases	0.10397
Bases in novel hairpin loops of all bases	0.15758
Bases in novel hairpin loops of all unpaired bases	0.44384
Bases in all hairpin loops of all bases	0.19450
Bases in all hairpin loops of all unpaired bases	0.54782
Bases in classic stems of all bases	0.06288
Bases in novel stems of all bases	0.58208
Bases in all stems of all bases	0.64496
H bonds in classic stems of all H bonds	0.06552
H bonds in novel stems of all H bonds	0.93448
H bonds in all stems of all H bonds	1.00000
Bases in bulges of all bases	0.10085
Bases in bulges of all unpaired bases	0.28404
Bases in interior loops of all bases	0.01491
Bases in interior loops of all unpaired bases	0.04198

Table 4: Expected lengths for different structural motifs.

Parameter	Expectation
Length of bridges	2.16110
Length of bond chains	2.77319
Length of cycles	3.16110
Length of runs of unpaired bases	1.32245
Length of segments in bases	1.89366
Length of classic hairpin loops	1.32245
Length of novel hairpin loops	1.32245
Length of all hairpin loops	1.32245
Length of classic stems in bases	2.12642
Length of novel stems in bases	2.67691
Length of all stems in bases	2.62657
Length of classic stems in H bonds	1.06321
Length of novel stems in H bonds	1.52618
Length of all stems in H bonds	1.48385
Length of bulges	1.32245
Length of interior loops	2.64490

Table 5: Comparison between expectations in the combinatorial model with and without base triples. Results for hairpin loops and bulges in classic structures are reproduced from Nebel (2002a), while those for stems, unpaired bases and H bonds are obtained from Hofacker et al. (1998).

Parameter	Quotient
Mean number of (all) hairpin loops	1.39311
Expected length of hairpin loops	0.81732
Mean number of (all) stems	1.36997
Expected length of stems in H bonds	1.26732
Mean number of bulges	0.44641
Expected length of bulges	0.81732
Mean number of unpaired bases	0.79387
Mean number of H bonds	1.73619

and longer stems – which might have been expected.

Our model is purely combinatorial, i.e. different structural motifs evolve uniformly and independent of any energy constraints. Thus, we do not expect it to behave realistically, i.e., to reflect the expected structural appearance of native RNA structures with base triples. Further theoretical research may use non-uniform stochastic models to narrow the gap. Nevertheless, in order to see if there are at least some parameters which agree and to provide the reader an insight into the real world behavior of the structural motifs studied here, we performed the following experiments. We downloaded structural data from the Protein Data Bank (PDB)[5] yielding 990 records. Each of the structures[6] has been analyzed using

[5] Found at http://www.rcsb.org/pdb/home/home.do, data downloaded June 6th 2013, search criteria *Everything* selecting polymertyp RNA.

[6] For our final statistical analysis, 3R1C has been erased for it showed a (presumably) flawed behavior, distorting the computed regressions.

RNAVIEW and a handmade program was used to count appearances of the various motifs. We had to omit 405 records since their structures contained pseudo-knots. For the resulting data on motif appearances, linear regressions were computed. The last column of Table 2 lists the obtained fittings (as a function of sequence length n), omitting constant additive terms. Details will be reported in the Supplement (link can be found in appendix).

3 Method and proofs

In this section, we explain how the asymptotics for the number of extended secondary structures and the diverse parameters were derived. We used the methodology which – in connection with RNA structure – has been introduced in Nebel (2002b). We will show detailed steps of the analysis for the number of extended secondary structures as well as for the number of hairpins. Since our other results are obtained in a very similar way, details on their derivation will be made available in the Appendix and a Supplement only.
We assume the reader to be familiar with generating functions and the $\mathcal{O}$-transfer method as well as with implicit algebraic functions and the Newton polygon method. For elaborate information on those, we refer the reader to Flajolet and Sedgewick (2009) and Hille (1962). Background information on these topics can also be found in Flajolet and Sedgewick (1993a,b, 2001).

3.1 Asymptotic number of extended secondary structures

We start with the unambiguous, pictorial grammar $\mathcal{G}$ for all extended secondary structures presented in (Höner zu Siederdissen et al., 2011, Fig. 3), translating it to a context-free grammar for the language of extended secondary structures as follows: Any pictorial intermediate symbol on the left-hand side of an equation in $\mathcal{G}$ is assigned a different intermediate symbol of our grammar. For example, we identify the axiom S with $\circ\!\!-\!\!\!-\!\!\circ$ from the first (leftmost) equation in $\mathcal{G}$. An isolated $\bullet$ for $\mathcal{G}$ is identified with $\ast$, a $\bullet$ connected by an arc either with (or) depending on the direction of the arc. Thus, the equation $\circ\!\!-\!\!\!-\!\!\circ = \bullet\circ\!\!-\!\!\!-\!\!\circ$ for example becomes production $S \to \ast S$ and the equation $\circ\!\!-\!\!\!-\!\!\circ = \bullet\overset{\frown}{\_}\bullet$ is translated into $S \to B$, B the intermediate introduced for pictogram $\bullet\overset{\frown}{\_}\bullet$. It gets only slightly more complicated to handle pictograms using symbols $\triangleleft$, $\triangleright$ and $\triangleleft\!\triangleright$. A symbol $\triangleleft\!\triangleright$ in $\mathcal{G}$ corresponds to a triple base, its kind depending on the arcs which are connected to the triangles. If an arc to the right (resp. left) is connected to the left (resp. right) triangle in $\triangleleft\!\triangleright$, then $\triangleleft\!\triangleright$ corresponds to ⸨ (resp. ⸩), otherwise $\triangleleft\!\triangleright$ stands for)(. The symbol at the other end of such an arc is translated accordingly, e.g. a $\bullet$ becomes an ordinary bracket. A single triangle $\triangleleft$ or $\triangleright$ corresponds to no bracket but represents a still missing corresponding bracket for an already produced symbol. In this way, the pictorial equation $\circ\!-\!\circ = \bullet\overset{\frown}{{}_{\circ-\triangleleft\triangleright}}\circ\!\!-\!\!\!-\!\!\circ$ of $\mathcal{G}$ contributes the production $S \to (L⸩S$ where L is the intermediate associated with $\circ\!-\!\triangleleft$ representing a substructure which must contribute a symbol corresponding to the so far unmatched bond of ⸩. Using three more nonterminal symbols R, C and D to identified with $\triangleright - \circ$, $\triangleright - \triangleleft$ and $\triangleright \times \triangleleft$ we finally arrive at the following context-free grammar (using | to separate alternatives for the same nonterminal) :

$$\begin{array}{lcl} S & \rightarrow & *S \mid * \mid BS \mid B \mid (L\rrparenthesis S \mid (L\rrparenthesis \mid \llparenthesis R)S \mid \llparenthesis R) \mid \llparenthesis D\rrparenthesis S \mid \llparenthesis D\rrparenthesis \mid (S)\!(R \mid \llparenthesis R)\!(R \\ L & \rightarrow & *L \mid BL \mid (L\rrparenthesis L \mid \llparenthesis R)L \mid \llparenthesis D\rrparenthesis L \mid (S)\!(C \mid (S \mid \llparenthesis R)\!(C \mid \llparenthesis R \\ R & \rightarrow & S)S \mid S) \mid L\rrparenthesis S \mid L\rrparenthesis \mid S)\!(R \\ C & \rightarrow & D \mid S \\ D & \rightarrow & S)L \mid S)\!(C \mid L\rrparenthesis L \\ B & \rightarrow & (S) \end{array} \tag{3.1}$$

It is easy to see that this grammar is also in agreement with our definition for extended secondary structures: either pairs of corresponding brackets are produced in a single step (like, e.g., for $B \rightarrow (S)$) or an unbalanced pair of brackets is generated where a partner corresponding to a base triple still is missing (e.g., $S \rightarrow (L\rrparenthesis$). For the latter the used nonterminal (L in case of $S \rightarrow (L\rrparenthesis$) signals a still missing bracket of appropriate type (left for nonterminal L). In case of $\llparenthesis D\rrparenthesis$ nonterminal D ensures that the two base triples will not be paired to each other twice.

In order to use this grammar to determine the total number of extended secondary structures of size n we translate it into a counting generating function. For $\mathcal{S}$ the class of all extended secondary structures we set $S(z) = \sum_{s\in\mathcal{S}} z^{|s|}$, $|s|$ the size (number of symbols resp. nucleotides) of structure s. In general, we assign every nonterminal symbol X the function $X(z)$, $X \in \{S, L, R, C, D, B\}$. Now it is well-known (see, e.g., (Flajolet and Sedgewick, 2001), (Flajolet and Sedgewick, 2009, I.5.4)) how to translate a context-free grammar into a system of equations for those functions; its solution for the function associated with the axiom (S in our case) provides the counting generating function for the language generated[7]. To derive that system, we make every rule with left-hand side X an additive term for the equation for $X(z)$, replacing terminal symbols by z (each terminal contributes to the size of a string generated thus must contribute to the exponent of z) and each nonterminal by its generating function; concatenation is replaced by multiplication. This way we arrive at

$$\begin{array}{rcl} S(z) & = & zS(z) + z + B(z)S(z) + B(z) + z^2L(z)S(z) + z^2L(z) + z^2R(z)S(z) + z^2R(z) \\ & & +z^2D(z)S(z) + z^2D(z) + z^2S(z)R(z) + z^2R(z)^2, \\ L(z) & = & zL(z) + B(z)L(z) + z^2L(z)^2 + z^2R(z)L(z) + z^2D(z)L(z) + z^2S(z)C(z) + zS(z) \\ & & +z^2R(z)C(z) + zR(z), \\ R(z) & = & zS(z)^2 + zS(z) + zL(z)S(z) + zL(z) + zS(z)R(z), \\ C(z) & = & D(z) + S(z), \\ D(z) & = & zS(z)L(z) + zS(z)C(z) + zL(z)^2, \\ B(z) & = & z^2S(z). \end{array}$$

The next step is to reduce the system of equations to a single equation for $S(z)$. This can be done by means of resultants or Groebner[8] bases and we find the (for the following steps

[7] Note that for an ambiguous grammar the resulting generating functions counts the number of strings together with their degree of ambiguity, i.e., together with the number of different ways a string can be derived.

[8] We used Wolfram Mathemematica's commands `GroebnerBasis` and `Resultant` to solve this task; for most of the subsequent steps we used Mathemematica, too. A corresponding notebook is available in the Supplement.

necessarily) irreducible equation[9] (abbreviating $S(z)$ by S)

$$\begin{aligned} S &= (4z^5)S^5 + (4z^3 - 7z^4 + 9z^5)S^4 + (-8z^2 + 11z^3 - 14z^4 + 7z^5)S^3 \\ & +(5z - 10z^2 + 14z^3 - 9z^4 + 2z^5)S^2 + (3z - 7z^2 + 7z^3 - 2z^4)S \\ & z - 2z^2 + z^3. \end{aligned} \tag{3.2}$$

Please note that at this point the connection of grammar and equation got lost. However, this is an inevitable consequence of the need for a single equation for our generating function. Connecting grammar and function at this point of the analysis asks for a grammar with a single nonterminal only. Obviously we cannot hope to easily find such a grammar. In the domain of generating functions however things become commutative (the order of terminals and intermediates on the right-hand side of a production is no longer of importance). Therefore it is possible to compute a corresponding single equation.

We are not able to solve this equation for S in order to get a closed form representation of $S(z)$. However, this is not a big problem as we can proceed along the following lines: Our final goal is to apply techniques from analytic combinatorics (the $\mathcal{O}$-transfer method, see (Flajolet and Sedgewick, 2009, VI.2)) to derive an exact asymptotic for the coefficient at z^n of $S(z)$ denoted $[z^n]S(z)$. To this end we need an expansion of $S(z)$ around its dominant singularity, i.e., its singularity of smallest modulus. This singularity and the corresponding expansion can be determined from the implicit representation of $S(z)$ given by (3.2) alone. For any algebraic function $S(z)$ implicitly given by a polynomial equation $F(z, S) = 0$ of degree k (in S) we normally have k distinct finite roots – the branches of the algebraic function S. There are values of z were this statement about the number of roots is not valid; the first being those for which the degree in S changes, i.e., for which the coefficient $p_k(z)$ of S^k in the polynomial equation becomes 0. There, at least one branch of the function disappears (at infinity) – thus being called a point at infinity – implying a pole (singularity) for one or more of the branches. In our cases the corresponding pole $z = 0$ cannot belong to the branch of interest to us for it would imply an infinite grows (of order 0^{-n}) for $[z^n]S(z)$. Furthermore, values of z where $F(z, S)$ has multiple roots coincide with the branch points of the algebraic function, i.e., points where two or more branches meet, and thus the function ceases to be analytic. According to (Hille, 1962, Theorem 12.2.1.) and (Flajolet and Sedgewick, 2009, Lemma VII.4) those can be determined by solving $D(z) = 0$, $D(z) := (-1)^{\frac{1}{2}k(k-1)}[p_k(z)]^{-1}R[F, F_w, w]$, $R[F, F_w, w]$ the resultant for F and the first derivative of F with respect to w; the roots of the equation $D(z) = 0$ provide a superset of the finite branch points. For our equation (3.2) we find

$$\begin{aligned} D(z) &= (1-z)^2 z^1 3(1+z)^8(64 - 560z + 1292z^2 + 1060z^3 - 8384z^4 + 13744z^5 - 7936z^6 \\ & -5135z^7 + 5850z^8 + 5279z^9 - 4984z^{10} - 1417z^{11} + 1342z^{12} + 205z^{13} - 200z^{14} \\ & +112z^{15}) \end{aligned}$$

with the smallest – we seek the dominant singularity – non-zero solution of $D(z) = 0$ being

$$z_0 := 0.2438282498191205681148525860 22\ldots \tag{3.3}$$

A close inspection of (3.2) shows that this is a branch point indeed (see Figure 9 for a graphical representation).

The next step is to compute the expansion of $S(z)$ around z_0. Since no closed form representation for $S(z)$ is available, we use the Newton polygon method for that task. We follow the

[9] We typically speak of irreducible polynomials however subtracting S from both sides of the equation yields a polynomial in variables z and S equated 0.

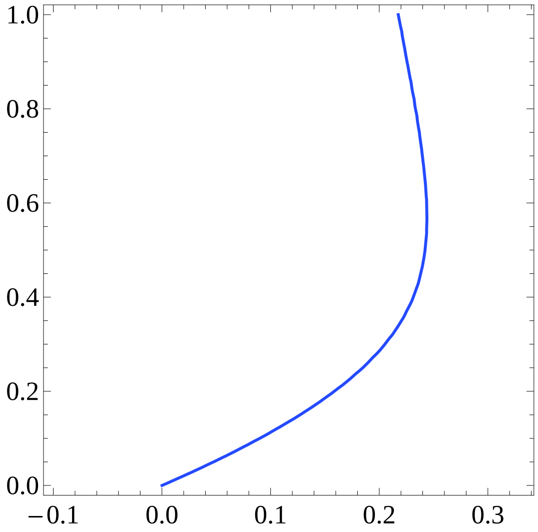


Figure 9: The implicit plot of our generating function $S(z)$ showing a branch point at $z = 0.243828\ldots$

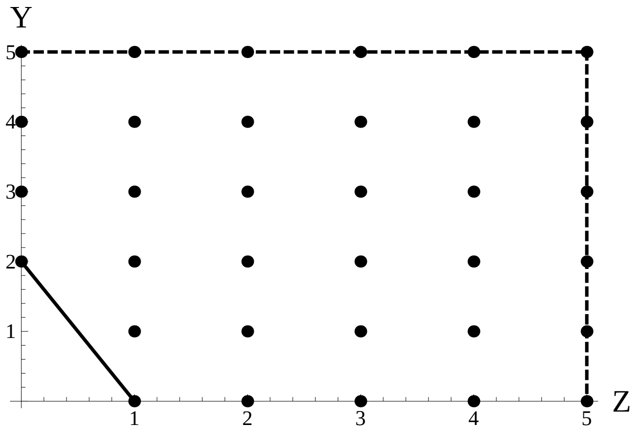


Figure 10: Newton diagram of $\hat{F}(Z,Y)$. The bold line depicts the least convex polygon of the Newton diagram.

approach discussed in (Hille, 1962, 12.3), shifting the branch point to the origin by means of the translations $z = z_0 - Z$ and $S = s_0 + Y$, s_0 the solution of (3.2) at z_0; we call the resulting polynomial equation $\hat{F}(Z,Y)$ (implicitly defining a function $Y(Z)$). We know a priori the existence of rational number α and complex number c such that $\lim_{Z\to 0} Z^{-\alpha}Y(Z) = c$ which provides the leading term of the needed expansion $Y(Z) \sim cZ^{\alpha}$. Now the Newton diagram allows to determine α. It has for each term Z^iY^j with a non-zero coefficient in polynomial $\hat{F}(Z,Y)$ a point (i,j) in the (Z,Y) plane and we construct the least convex polygon containing these points. Figure 10 shows the diagram for $\hat{F}$.

The negative inverse slope of the (in our case unbroken) line connecting the points of smallest modulus on the two axes equals α, thus $\alpha = \frac{1}{2}$ holds in our case[10]. Knowing α, we obtain c by solving $\hat{F}(Z, cZ^{1/2})$ and subsequently setting $Z = 0$. This way we arrive at the following

[10] In general, each line segment composing the connection of the two axes constitutes a possible exponent α, see (Flajolet and Sedgewick, 2009, Sec. VII.7.1.).

leading term of the expansion

$$c := -1.7985947826610932952573739398 12\ldots$$

In order to compute more terms, the whole process could be repeated after eliminating the main term from Y. However, for our purposes there is no need to compute additional terms and we continue our analysis based on

$$Y(Z) \sim c\sqrt{Z} + \mathcal{O}(Z).$$

In a third and last step, we apply the $\mathcal{O}$-transfer method (see (Flajolet and Sedgewick, 2009, VI.2)) in order to determine an asymptotic for the nth coefficients from the series expansion just computed. Keeping in mind the substitution $Z = z_0 - z$, we arrive at

$$-\frac{c\sqrt{z_0}}{2\sqrt{\pi n^3}}\, z_0^{-n} + \mathcal{O}(a^{-n}n^{-5/2}).$$

Equation (2.1) shows a rounded version of this result.

3.2 Asymptotic number of hairpins

The general methodology to analyze the number and length of the various structural motifs present in extended secondary structures is always the same. Starting with an unambiguous context-free grammar for the language of extended structures we derive a generating function and compute its coefficient. Different from what we did in the last section we now need a secondary variable y to mark the occurrence of a motif or all its parts in case we want to analyze lengths (see (Flajolet and Sedgewick, 2009, Chap. III) for details). Even if this task appears rather mechanic, one detail needs to be considered. Any structural motif resp. the symbols contributing to its length need to be uniquely associated to one or several of the productions of the grammar used such that the application of the rule(s) is in one-to-one correspondence to the motif showing up in the structure resp. a symbol contributing to the motif in question.

The grammar of before section is not well-designed to this end and we will make use of the grammar shown in Table 6 – denoted G in the sequel – and variants thereof to analyze the various motifs. Grammar G uses nonterminal S to generate any extended secondary (sub-)structure as a sequence of segments (bond chain or unpaired nucleotide). Any bond chain is produced from intermediate H. Here some H rules generate brackets that are corresponding (e.g. $H \to \boldsymbol{(}S\boldsymbol{)}$ or $H \to$ ⦅RTL⦆ where the *outer* brackets are corresponding), some rules just possess a nonterminal that will produce a corresponding bracket in a later step (e.g. $H \to$ ⦅$RTWTL$⦆ where the *outer* bracket of ⦅ will correspond with a χ resulting from W and the *inner* bracket of ⦅ will find its partner in the string derived from R). Accordingly, nonterminal L (resp. R) is used to produce left (resp. right) brackets that are needed to correspond with symbols already generated. Intermediate W is used to produce the run of symbols χ within bridges. We could prove G's equivalence to grammar (3.1) by providing a sequence of rule transformations which step by step changes the set of productions from one to the other – and this indeed was the way we constructed G. On this way one can easily check that each modification does not change the language generated by simulation arguments, ambiguity issues could be addressed using generating functions (details on this way of reasoning can be found later in this section and in the Appendix, where similar arguments are used for variants of G). However, we decided to provide a more elegant proof here: First we proof by induction that every string generated by G encodes an extended

Table 6: Production rules of an unambiguous, context-free grammar G for extended RNA secondary structures with base triples. Capital letters are used for non-terminal symbol, while unpaired (*), pair (‘(’, ‘)’) and triple bases (‘⦅’, ‘⋈’, ‘⦆’) constitute the terminal symbols.

S	$\rightarrow$	$* \mid *S \mid H \mid HS$
H	$\rightarrow$	⦅$RTWTL$⦆ $\mid$ ⦅RTL⦆
		$\mid$ ⦅SWS⦆ $\mid$ (S) $\mid$ (SWS)
		$\mid$ ⦅$RTWS$) $\mid$ ($SWTL$⦆
		$\mid$ ⦅RT) $\mid$ (TL⦆
L	$\rightarrow$	⦅$RTWS$ $\mid$ (SWS $\mid$ (S $\mid$ ⦅RT
R	$\rightarrow$	$SWTL$⦆ $\mid$ SWS) $\mid$ S) $\mid$ TL⦆
T	$\rightarrow$	$\epsilon \mid S$
W	$\rightarrow$	⋈SW $\mid$ ⋈

secondary structure according to Definition 1, i.e., is an element of $\mathcal{S}$. Second, we show that the grammar is unambiguous. By observing that (3.1) and G have the same counting generating functions we conclude that G is equivalent to grammar (3.1).

For nonterminal X define $\mathcal{L}_X^N$ to be the language that could be derived from X using at most N steps in a leftmost derivation and $\mathcal{L}_X := \lim_{N\to\infty} \mathcal{L}_X^N$, $X \in \{S, H, L, R, T, W\}$. First we observe that ⋈ and)(can be assume equivalent with respect to δ from Definition 1 and thus we will identify the two in the proof to prevent unnecessary case distinctions. We show by induction on the number of steps in a leftmost derivation that

$$\mathcal{L}_S \equiv_\delta \varepsilon,\ \mathcal{L}_H \equiv_\delta \varepsilon,\ \mathcal{L}_L \equiv_\delta \texttt{(},\ \mathcal{L}_R \equiv_\delta \texttt{)},\ \mathcal{L}_T \equiv_\delta \varepsilon,\ \mathcal{L}_W \equiv_\delta \texttt{)(},$$

writing $\mathcal{L}_X \equiv_\delta \rho$ as a short form for $(\forall w \in \mathcal{L}_X)(w \equiv_\delta \rho)$.

We start with considering derivations of at most 2 steps. For S we can generate * and ** but no other word since H needs at least two steps to generate a terminal string; obviously, $\{*, **\} \equiv_\delta \varepsilon$ holds. For the other nonterminals we find $\mathcal{L}_H^2 = \{\texttt{(}*\texttt{)}\} \equiv_\delta \varepsilon$, $\mathcal{L}_L^2 = \{\texttt{(}*\} \equiv_\delta \texttt{(}$, $\mathcal{L}_R^2 = \{*\texttt{)}\} \equiv_\delta \texttt{)}$, $\mathcal{L}_T = \{\varepsilon, *\} \equiv_\delta \varepsilon$ and[11] $\mathcal{L}_W^2 = \{\texttt{)(}\} \equiv_\delta \texttt{)(}$ and the anchor follows. Now assume the claim holds for derivations of at most N steps and consider derivations of length at most $N+1$. For S we have 4 alternatives, and we can apply the claim to any symbol showing up on the right-hand side after a rule was used. Since any string derived from S or H in at most N steps is equivalent ε we conclude $\mathcal{L}_S^{N+1} \equiv_\delta \varepsilon$. Regard $H \rightarrow$ ⦅$RTWTL$⦆ as a first step. By the induction hypothesis everything which might be derived from R (resp. T, W, L) can be reduce to) (resp. ε,)(, (). Thus after the corresponding reductions have been performed inside-out we are left with ⦅))((⦆ which obviously reduces to ε applying the equations from δ. It is an easy exercise to show that the same holds when applying one of the other alternatives for H as a first derivation step, such that $\mathcal{L}_H^{N+1} \equiv_\delta \varepsilon$ follows. Consider $L \rightarrow$ ⦅$RTWS$ as the first derivation step. Again, we can make use of the hypothesis to reduce R, T, W and S getting ⦅))(which obviously reduces to (. At this point it should be obvious how to proceed until the induction step is entirely proven. We remark while passing that by inspection of the rules for nonterminal H it is easy to show that G cannot generate any pair of corresponding brackets without at least one symbol in between (thus honoring condition $w \equiv_{\delta \setminus \{* \equiv_\delta \varepsilon\}} \rho \implies \rho = w$ of Definition 1).

To show that G is unambiguous, note that we can decompose any word derived from S into a run of * followed by a pair of brackets (generated from H) and potentially a suffix generated from S again. The leading symbols * cannot imply ambiguities for their number

[11] Here we make use of before mention equivalence ⋈ ≡)(.

Table 7: Grammar G_{ch} used for counting number and total length of classic hairpin loops.

S	$\rightarrow$	$\texttt{*} \mid \texttt{*}S \mid H \mid HS$
H	$\rightarrow$	$\llparenthesis RTWTL \rrparenthesis \mid \llparenthesis RTL \rrparenthesis$
		$\mid \llparenthesis SWS \rrparenthesis \mid \texttt{(}V\texttt{)} \mid \texttt{(}SWS\texttt{)}$
		$\mid \llparenthesis RTWS\texttt{)} \mid \texttt{(}SWTL \rrparenthesis$
		$\mid \llparenthesis RT\texttt{)} \mid \texttt{(}TL \rrparenthesis$
L	$\rightarrow$	$\llparenthesis RTWS \mid \texttt{(}SWS \mid \texttt{(}S \mid \llparenthesis RT$
R	$\rightarrow$	$SWTL \rrparenthesis \mid SWS\texttt{)} \mid S\texttt{)} \mid TL \rrparenthesis$
T	$\rightarrow$	$\epsilon \mid S$
W	$\rightarrow$	$\texttt{)(}SW \mid \texttt{)(}$
V	$\rightarrow$	$P \mid Q$
P	$\rightarrow$	$\texttt{*} \mid \texttt{*}P$
Q	$\rightarrow$	$\texttt{*}Q \mid H \mid HS$

uniquely determines which rules to apply. For any combination of the outermost brackets in $\{\texttt{(}, \llparenthesis\} \times \{\texttt{)}, \rrparenthesis\}$ G has several alternatives to generate them. However, we can easily decide which has to be applied by considering the respective corresponding brackets. For example, the outermost pair $\llparenthesis\ \rrparenthesis$ may not be corresponding at all; then $\llparenthesis RTWTL \rrparenthesis$ must be used such that $\llparenthesis$ (resp. $\rrparenthesis$) becomes corresponding with the residue of the string produced from R (resp. L) and a symbol $\texttt{)(}$ introduced by W. If $\llparenthesis\ \rrparenthesis$ are corresponding with each other as well as with the residues of the string produced from R resp. L, rule $H \rightarrow \llparenthesis RTL \rrparenthesis$ must be used. Finally, if the two are corresponding with each other and with an interior $\texttt{)(}$, $H \rightarrow \texttt{(}SWS\texttt{)}$ is the choice. It is now an easy exercise, to use the same kind of reasoning to see that the outside-in decomposition of any terminal string generated by G always allows the unique identification of the next production to use. Finally please note that after translating G into a system of generating functions and reducing this to a single equation in S and z, we rediscover (3.2). This proves G to unambiguously[12] generate $\mathcal{S}$.

Now let us try to count the number of hairpins in the extended secondary structures of size n. We first observe that any hairpin loop uniquely corresponds to a hairpin such that counting hairpin loops is sufficient. Here we want to distinguish classic from novel loops. According to Definition 3 a classic hairpin loop of length l is given by a substring $\texttt{(*}^l\texttt{)}$, $l \geq 1$. Within G, a pair of corresponding brackets $\texttt{()}$ is generated by production $H \rightarrow \texttt{(}S\texttt{)}$. Unfortunately, S now can either generate a hairpin loop or a structure with further H bonds and accordingly we cannot identify the number of rule applications of $H \rightarrow \texttt{(}S\texttt{)}$ and the number of classic hairpins. However, introducing a new nonterminal V which either can generate $\{\texttt{*}\}^+$ (production $V \rightarrow P$, P a second new nonterminal with $P \rightarrow \texttt{*} \mid \texttt{*}P$) or $\mathcal{L}_S \setminus \{\texttt{*}\}^+$ (production $V \rightarrow Q$, Q a third new nonterminal with $Q \rightarrow \texttt{*}Q \mid H \mid HS$) and replacing $H \rightarrow \texttt{(}S\texttt{)}$ by $H \rightarrow \texttt{(}V\texttt{)}$ makes it possible to identify a classic hairpin with the application of rule $V \rightarrow P$ and to count their length by taking[13] $P \rightarrow \texttt{*}P$ into account, too. Obviously this modification does not change the generated language and no ambiguities are introduced. Table 7 shows the complete resulting grammar G_{ch}. To determine the total number of all classic hairpins in all extended secondary structures of size n we translate the productions of G_{ch} in the same way into a system of equations as before with the only difference that now a second variable y is used to mark the application of $V \rightarrow P$, assuming

[12] To show that G only produces strings in $\mathcal{S}$ and that the generating functions coincide is not sufficient because missing extended secondary structures in general could have been balanced in number by ambiguities.

[13] After production $f : V \rightarrow P$ has been used, $P \rightarrow \texttt{*}$ is applied exactly one. Thus counting the number of applications of rule f is quantitatively equivalent to count applications of rule $P \rightarrow \texttt{*}$.

all generating functions to be bivariate (depending on variables z and y). Accordingly, $V \to P$ is translated into $V(z, y) = yP(z, y)$ giving rise to an increment of the exponent of y every time this rule is applied. Afterwards, the resulting system of equations is reduced into a single one in S, z and y using standard techniques. Then, we take the implicit partial derivative with respect to y and set $y = 1$ afterwards (see (Flajolet and Sedgewick, 2009, Chap. III)). This way, we get any implicit equation for the resulting generating function $S'(z, 1)$ whose coefficient at z^n is the total number of classic hairpins in all structures of size n. This equation is in two symbols, the derivative $S'(z, 1)$ with respect to y at $y = 1$ and $S(z, 1) = S(z)$, i.e., the generating function counting all extended secondary structures. For the latter we use our equation from the analysis before and reduce the resulting system of two equations to get an implicit equation in two symbols S' and z for the generating function in question. At this point we can compute the corresponding coefficient asymptotic (by Newton polygon method and $\mathcal{O}$-transfer as before) which then is divide by the asymptotic number of extended secondary structures of size n to compute the average number of hairpin loops in an extended structures of size n. Since the number of hairpins in a structure of size n is upper bounded by n, the coefficient of generating function $S'(z, 1)$ can at most be a factor n larger than that of generating function $S(z)$. Therefore, the dominant singularity must be the same since the exponential rate of growth cannot change – an observation which holds for all the parameters considered. We only provide the key results of the remaining analysis. The implicit representation of $S'(z, 1) =: T$ is given by

$$T^5(112z^{18} - 536z^{17} + 1141z^{16} + 15z^{15} - 4628z^{14} + 3088z^{13} + 14638z^{12} - 23522z^{11} - 1864z^{10} +19740z^9 + 16297z^8 - 68289z^7 + 75380z^6 - 40784z^5 + 7128z^4 + 4560z^3 - 3164z^2 + 752z - 64) + T^4(112z^{18} - 424z^{17} + 717z^{16} + 732z^{15} - 3896z^{14} - 808z^{13} + 13830z^{12} - 9692z^{11} - 11556z^{10} + 8184z^9 +24481z^8 - 43808z^7 + 31572z^6 - 9212z^5 - 2084z^4 + 2476z^3 - 688z^2 + 64z) + T^3(50z^{18} - 122z^{17} +162z^{16} + 604z^{15} - 1225z^{14} - 2013z^{13} + 4203z^{12} + 2069z^{11} - 4943z^{10} - 3621z^9 + 9134z^8 - 5546z^7 + 1367z^6 - 119z^5) + T^2(11z^{18} - 8z^{17} + 34z^{16} + 172z^{15} - 101z^{14} - 526z^{13} + 139z^{12} + 730z^{11} - 6z^{10} -758z^9 + 442z^8 - 64z^7 - z^6) + T(z^{18} + z^{17} + 10z^{16} + 34z^{15} + 23z^{14} - 27z^{13} - 36z^{12} - 8z^{11} + 2z^{10}) + z^{16} + 4z^{15} + 6z^{14} + 4z^{13} + z^{12}.$$

Note that for T our dominant singularity is a point at infinity, making the coefficient of T^5 to become zero. However, we still can apply the Newton polygon method substituting T by $1/U$, multiplying with U^5 afterwards. This way, the point at infinity is translated into a branch point at z_0 for the resulting implicit equation for $U(z)$. Proceeding as before, i.e., shifting the branch point to the origin (substitution $Z = z_0 - z$), deriving the Newton diagram etc. finally yields the following leading term of the expansion

$$U(z) \sim 163.375126468432414746\ldots\sqrt{Z}.$$

We have to first undo the substitution $Z = z_0 - z$ and then make use of $T = 1/U$. This way we find

$$T(z) \sim \frac{0.00612088279052210245308374699 5\ldots}{\sqrt{z_0}\sqrt{1 - \frac{z}{z_0}}}$$

and the $\mathcal{O}$-transfer implies

$$[z^n]T(z) \sim \frac{0.006120882790522102453083746995\ldots}{\sqrt{z_0}\sqrt{\pi n}} z_0^{-n}.$$

Computing the quotient of before asymptotic and the asymptotic number of extended structures of size n yields an average number of $0.027914\ldots n$ classic hairpin loops in an extended

Table 8: Grammar G_{nh} used for counting the number and total length of novel hairpin loops.

S	$\to$	$* \mid *S \mid H \mid HS$
H	$\to$	$⦅RTWTL⦆ \mid ⦅RTL⦆$
		$\mid ⦅VWV⦆ \mid (S) \mid (VWV)$
		$\mid ⦅RTWV) \mid (VWTL⦆$
		$\mid ⦅RT) \mid (TL⦆$
L	$\to$	$⦅RTWV \mid (VWV \mid (V \mid ⦅RT$
R	$\to$	$VWTL⦆ \mid VWV) \mid V) \mid TL⦆$
T	$\to$	$\epsilon \mid S$
W	$\to$	$)(VW \mid)($
V	$\to$	$P \mid Q$
P	$\to$	$* \mid *P$
Q	$\to$	$*Q \mid H \mid HS$

structure of size n (see Table 2). We already explained above, how to label productions of G_{ch} by y in order to count the total length of all classic hairpins. Dividing the resulting asymptotic number – its computation follows exactly the same lines and is detailed in the Supplement – by the number of hairpin loops just computed yields the expected length of a single classic hairpin loop of $1.32245\ldots$ as reported in Table 4. Dividing the total length of all classic hairpin loops by the number of extended secondary structures provides the *Number of bases in classic hairpin loops* $0.03692\ldots n$ as given in Table 2. Finally, dividing this number by n (resp. by the expected number of unpaired bases $0.35504\ldots n$) yields the corresponding proportion *Bases in classic hairpin loops of all bases* (resp. *Bases in classic hairpin loops of all unpaired bases*) as reported in Table 3. Again, details on the corresponding analysis can be found in the Supplement.

To count the novel hairpins, we need to apply similar changes to grammar G at different places (rules). In detail, we have to identify those positions in the right-hand sides of productions with premise H that might produce a pair of corresponding brackets different from (). Here, we will have the possibility to either generate a run of symbols * or a structure with additional paired bases, the first alternative implying a novel hairpin loop. Table 8 shows the resulting rule set where nonterminals V, P and Q are used in the same way as before. We will not discuss all the changes applied in detail but demonstrate the way of reasoning using the example of production $H \to (VWTL⦆$. This rule results from $f : H \to (SWTL⦆$ for G. We first observe that the bracket (produced by f will be in correspondence with)(generated from W since S (as proven above) can only produce strings that reduce to ε. Accordingly, (together with)(will imply a hairpin loop iff S generates a run of *. Thus, replacing S by V allows to keep track of those loops. Furthermore, ⦆ might contribute a second novel hairpin loop. However, we need to consult the rule applied for L to handle this one. If for instance $L \to (S$ is applied, this might yield a hairpin, thus we replace S by V for this production. Analogously, $g : L \to (SWS$ might give rise of two novel hairpin loops, one between (produced by g and)(resulting from W and one between)(and ⦆ stemming from f. As a consequence, both occurrences of S in g are replaced by V. Considering all the other possibilities in the very same way finally yields grammar G_{nh} as shown in Table 8.

From this point on, the analysis follows the very same lines: we translate the grammar into a system of generating function, marking terminal symbols by z and the application of rule $V \to P$ by y thus counting the total number of novel hairpins by taking the derivative

with respect y, setting $y = 1$ afterwards. If we additionally mark $P \to \ast P$ by y we keep track of the total length of all novel hairpin loops. Computing an expansion of the resulting generating function(s) at our dominant singularity finally provides a precise asymptotic for the number(s) in question. It should be obvious how to derive the related averages and proportions reported in Tables 2, 3 and 4.
We close this section my noting that the analysis of all hairpin loops can either be performed by adding up the results obtained for classic and novel hairpin loops or by modifying grammar G_{nh}, replacing $H \to \texttt{(}S\texttt{)}$ by $H \to \texttt{(}V\texttt{)}$.

4 Conclusion

In this paper, we examined an extended combinatorial model of RNA secondary structures considering base triples. Given the fact that the sequence of the first exact numbers of structures is not contained in the The On-Line Encyclopedia of Integer Sequences database, extended RNA secondary structures with base triples can be considered a new combinatorial object not in direct bijection to any combinatorial class studied before.
After providing precise definitions for the structures themselves as well as for various structural motifs capable of describing their overall appearance, we have presented a precise asymptotic for the number of extended RNA secondary structures as a function of their size n. This way we proved that introducing base triples increases the number of possible foldings exponentially. This gain has considerable effects on practical applications such as folding algorithms in terms of search space size and runtime. Second, we have shown averages, proportions and lengths of various structural motifs including both classic and novel ones. Also, we have pointed out differences in expectations compared to both classic models and nature. For the latter, we used data from the Protein Data Bank to derive statistics on structural parameters of native RNA molecules with base triples.

Future research based on our model could make use of stochastic context-free grammars in order to bring the model closer to nature. A further extension of the model by introducing pseudoknots might be another worthwhile effort.

Author disclosure statement

No competing financial interests exist.

Acknowledgments

We thank the anonymous referees for valuable comments on a first version of that paper which helped a lot to improve our presentation.

References

Abu Almakarem, A. S., Petrov, A. I., Stombaugh, J., Zirbel, C. L., and Leontis, N. B. (2012). Comprehensive survey and geometric classification of base triples in RNA structures. *Nucleic Acids Research*, 40(4):1407–1423.

Bousquet-Mélou, M. and Xin, G. (2006). On partitions avoiding 3-crossings. *Séminaire Lotharingien de Combinatoire*, 54:B54e.

Chen, W. Y., Qin, J., and Reidys, C. M. (2008). Crossings and Nestings in Tangled Diagrams. *The Electronic Journal of Combinatorics*, 15(1):R86.

Conn, G. L., Gutell, R. R., and Draper, D. E. (1998). A Functional Ribosomal RNA Tertiary Structure Involves a Base Triple Interaction. *Biochemistry*, 37(34):11980–11988.

Doherty, E. A., Batey, R. T., Masquida, B., and Doudna, J. A. (2001). A universal mode of helix packing in RNA. *Nat Struct Biol*, 8(4):339–43.

Flajolet, P. and Sedgewick, R. (1993a). The Average case analysis of algorithms: complex asymptotics and generating functions. Rapport de recherche RR-2026, INRIA.

Flajolet, P. and Sedgewick, R. (1993b). The Average case analysis of algorithms: Counting and generating functions. Rapport de recherche RR-1888, INRIA.

Flajolet, P. and Sedgewick, R. (2001). Analytic combinatorics: Functional Equations, Rational and Algebraic Functions. Rapport de recherche RR-4103, INRIA.

Flajolet, P. and Sedgewick, R. (2009). *Analytic Combinatorics*. Cambridge.

Harrison, M. A. (1978). *Introduction to Formal Language Theory*. Addison-Wesley Longman Publishing Co.

Hille, E. (1962). *Analytic Function Theory*, volume 2. Blaisdell Publishing Company.

Hofacker, I. L. (2003). Vienna RNA secondary structure server. *Nucleic Acids Research*, 31(13):3429–3431.

Hofacker, I. L., Schuster, P., and Stadler, P. F. (1998). Combinatorics of RNA secondary structures. *Discrete Applied Mathematics*, 88(1-3):207–237.

Höner zu Siederdissen, C., Bernhart, S. H., Stadler, P. F., and Hofacker, I. L. (2011). A folding algorithm for extended RNA secondary structures. *Bioinformatics*, 27(13):129–136.

Hopcroft, J. E., Motwani, R., and Ullman, J. D. (1979). *Introduction to Automata Theory, Languages, and Computation*, volume 2. Addison-wesley Reading, MA.

Kemp, R. (1996). On the Average Minimal Prefix-Length of the Generalized Semi-Dycklanguage. *RAIRO Theoretical Informatics and Applications*, 30(6):545–561.

Knudsen, B. and Hein, J. (1999). RNA secondary structure prediction using stochastic context-free grammars and evolutionary history. *Bioinformatics*, 15(6):446–454.

Leontis, N. B. and Westhof, E. (2001). Geometric nomenclature and classification of RNA base pairs. *RNA*, 7(4):499–512.

Nebel, M. E. (2002a). Combinatorial Properties of RNA Secondary Structures. *Journal of Computational Biology*, 9:541–573.

Nebel, M. E. (2002b). On a Statistical Filter for RNA Secondary Structures. Frankfurter Informatik-Berichte 5/02 (technical report), Johann Wolfgang Goethe-Universitaet.

Nebel, M. E. (2004). Investigation of the Bernoulli-Model of RNA Secondary Structures. *Bulletinof Mathematical Biology*, 66:925–964.

Nussinov, R., Pieczenik, G., Griggs, J. R., and Kleitman, D. J. (1978). Algorithms for Loop Matchings. *SIAM Journal on Applied Mathematics*, 35(1):pp. 68–82.

Parisien, M. and Major, F. (2008). The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data. *Nature*, 452(5):51–5.

Pipas, J. M. and McMahon, J. E. (1975). Method for Predicting RNA Secondary Structure. *Proceedings of the National Academy of Sciences of the United States of America*, 72(6):2017–2021.

Qin, J. and Reidys, C. M. (2007). A combinatorial framework for RNA tertiary interaction. Technical Report 0710.3523, arXiv.

Sankoff, D. and Kruskal, J. B. (1999). *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison]*. CSLI Publ.

Stein, P. R. and Waterman, M. S. (1979). On some new sequences generalizing the Catalan and Motzkin numbers. *Discrete Mathematics*, 26(3):261 – 272.

Waterman, M. S. (1978). Secondary Structure of Single-Stranded Nucleic Acids. In *Studies on foundations and combinatorics, Advances in mathematics supplementary studies, Academic Press N.Y., 1:167 – 212*, pages 167–212.

Zuker, M. and Stiegler, P. (1981). Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Research*, 9(1):133–148.

5 Appendix

In this appendix, we discuss in detail the modifications to grammar G needed to analyze the parameters not discussed in Section 3. The corresponding computations are presented in the Supplement (http://wwwagak.cs.uni-kl.de/downloads/papers/BaseTriplesSupplement.pdf – PDF version – or http://wwwagak.cs.uni-kl.de/downloads/papers/BaseTriplesSupplement.nb – Mathematica notebook).

Most of the parameters could be examined without altering grammar G (see Table 6). Nevertheless, some parameters (e.g. novel stems) require extensive modifications in order to distinguish them from other parts of the extended secondary structures. These modifications include doubling or restructuring parts of the rule set, but were done without introducing unambiguity and keeping equivalence to the original grammar in mind. The restructuring of grammar G in the analysis of hairpin loops is a representative example.

The following table lists the various structural motifs together with the name of the grammar and rule(s) used for their analysis. If not stated otherwise, number and length of the motifs are considered on basis of the same grammar and thus are not distinguished. The listed productions must be marked by y or y^2, depending on their contribution to the parameter. In cases where this distinction is complicated, the marking is provided, too. The corresponding grammars and their construction are detailed afterwards.

Parameter	CFG	rule(s) used to access parameter (marking by y)
unpaired bases	G	$S \to *$, $S \to *S$
runs of unpaired bases	G_{ru}	$D \to *$
H bonds, total	G	any rule producing a bracket symbol excluding $H \to \llparenthesis RTWTL \rrparenthesis$, $H \to (SWS)$, $H \to \llparenthesis RTWS)$, $H \to (SWTL\rrparenthesis$, $L \to \llparenthesis RTWS$, $L \to (SWS$ and the analogous R-productions since for all those rules H bonds for the produced bracket symbols are already counted elsewhere (e.g. H bonds for (SWS) are counted by W production)
H bonds in simple pairs	G	$H \to (S)$
H bonds in base triples	G	same as for total number H bonds excluding $H \to (S)$
paired bases with 1 H bond, total	G	every rule producing (and/or)
paired bases with 1 H bond in simple pairs	G	$H \to (S)$
paired bases with 1 H bond outside simple pairs	G	every rule produce (and/or) except $H \to (S)$
paired bases with 2 H bond, total	G	any rule producing $\llparenthesis$, $\rrparenthesis$ or $\between$
paired bases with 2 H bond, center position	G	any rule producing $\between$
number cycles	G	$H \to \llparenthesis SWS \rrparenthesis$
length of cycles	G_{lc}	both V productions
bridges	G	$W \to \between$ for number, both W rules for length (in H bonds)
bond chains	G	any H production
segments	G	any S production
classic hairpin loop	G_{ch}	$P \to *$ for number, both P rules for length
novel hairpin loops	G_{nh}	$P \to *$ for number, both P rules for length
classic stems	G_s	$H \to (H_3)$ for number, $H \to (H_3)$, $H_3 \to (H_3)$ for size
novel stems	G_s	$H \to H_2$, $H \to (H_4)$, $L_1 \to (V$, $L_1 \to \llparenthesis R_2$, $L_1 \to \llparenthesis R_1S$, $R_1 \to V)$, $R_1 \to L_2\rrparenthesis$, $R_1 \to SL_1\rrparenthesis$, $W \to \between VW$, $W \to \between$ for number, all H productions but $H \to (H_3)$, $H_3 \to H_1$, all H_4 productions, $V \to (V)$, $V \to H_1$, $V \to H_2$, all L_1, L_2, R_1 and R_2 productions, all W productions and (only when counting multiple affiliations) $H_2 \to \llparenthesis VW \rrparenthesis$ for length
bulges	G_b	any rule having D_2 in its conclusion for number, both D_2 productions for length
interior loops	G_{il}	$V \to D_2HD_2$ for number, both D_2 productions for length

Runs of unpaired nucleotides

Starting with G we cannot decide whether the application of $S \to *$ or $S \to *S$ can uniquely be identified with a run of unpaired bases. The first rule could be used to identify runs longer than 1 symbol since then it must be applied to terminate a growing sequence of $*$. However, if the run consists only of a single symbol then the applications of the second rule

Table 10: Grammar G_{ru} used for counting the number of runs of unpaired bases.

S	$\rightarrow$	$D \mid DH \mid DHS \mid H \mid HS$
H	$\rightarrow$	$\llparenthesis RTWTL \rrparenthesis \mid \llparenthesis RTL \rrparenthesis$
		$\mid \llparenthesis SWS \rrparenthesis \mid (S) \mid (SWS)$
		$\mid \llparenthesis RTWS) \mid (SWTL \rrparenthesis$
		$\mid \llparenthesis RT) \mid (TL \rrparenthesis$
L	$\rightarrow$	$\llparenthesis RTWS \mid (SWS \mid (S \mid \llparenthesis RT$
R	$\rightarrow$	$SWTL \rrparenthesis \mid SWS) \mid S) \mid TL \rrparenthesis$
D	$\rightarrow$	$* \mid * D$
T	$\rightarrow$	$\epsilon \mid S$
W	$\rightarrow$	$)(SW \mid)($

need to be counted for the produced S might be replace by H in the next step, implying the run to be terminated. Unfortunately, counting both would yield the length and not the number of runs. Therefore we introduce a new nonterminal D to stand for a non-empty run for unpaired nucleotides $*$ to be generated. Thus we have productions $D \rightarrow * \mid *D$. Furthermore, the set of rules for S is replace by $S \rightarrow D$ (the structure generated is entirely unpaired), $S \rightarrow DH$ (the structure starts with a run of $*$ which is followed by an outermost pair of brackets its closing bracket being the last symbol of the structure), $S \rightarrow DHS$ (the structure starts with a run of $*$ which is followed by an outermost pair of brackets and at least one more symbol), $S \rightarrow H$ (the structure consists of an outermost pair of brackets which constitute the first and last symbol of the generated encoding) and $S \rightarrow HS$ (the structure starts with an outermost pair of brackets followed by at least one more symbol). By "the structure" in before discussion we mean the encoding generated from intermediate S which is not necessarily the entire encoding of an extended secondary structure but might be a substring of one. It is obvious that the different cases for the productions introduced for S are disjoint and complete, thus the language generated is not changed and the grammar remains unambiguous. Table 10 shows the resulting complete grammar.

Length of cycles

It is easy to analyze the number and length of bridges, since the application of rule $W \rightarrow)($ is in one-to-one correspondence with the generation of a run of corresponding $)($ while such a run always belongs to exactly one bridge. Furthermore, each application of $W \rightarrow)(SW$ contributes to a bridge's length. The same holds for the number of cylces where production $H \rightarrow \llparenthesis SWS \rrparenthesis$ is the only one to initiate one. However, the length of cycles cannot be analyzed without modifying grammar G since we cannot distinguish whether the application of rule $W \rightarrow)(SW$ increases an arbitrary bridge or a bridge being a cycle. For this reason, we introduce nonterminal V, assumed to be a copy of W (thus having the productions $V \rightarrow)(SV \mid)($) and replace W in the right-hand side of $H \rightarrow \llparenthesis SWS \rrparenthesis$ by V. Obviously the language generated has not been changed and no ambiguities have been introduced; the V productions can be used to count the length of cycles. The resulting grammar G_{lc} is shown in Table 11.

Table 11: Grammar G_{lc} used for counting the total length of cycles.

S	$\rightarrow$	$\bullet \mid \bullet\, S \mid H \mid HS$
H	$\rightarrow$	$\llparenthesis RTWTL \rrparenthesis \mid \llparenthesis RTL \rrparenthesis$
		$\mid \llparenthesis SVS \rrparenthesis \mid (S) \mid (SWS)$
		$\mid \llparenthesis RTWS) \mid (SWTL \rrparenthesis$
		$\mid \llparenthesis RT) \mid (TL \rrparenthesis$
L	$\rightarrow$	$\llparenthesis RTWS \mid (SWS \mid (S \mid \llparenthesis RT$
R	$\rightarrow$	$SWTL \rrparenthesis \mid SWS) \mid S) \mid TL \rrparenthesis$
T	$\rightarrow$	$\epsilon \mid S$
V	$\rightarrow$	$\boldsymbol{)(} SV \mid \boldsymbol{)(}$
W	$\rightarrow$	$\boldsymbol{)(} SW \mid \boldsymbol{)(}$

Table 12: Grammar G_s used for counting the number and total length of classic, novel and all stems.

S	$\rightarrow$	$* \mid *S \mid H \mid HS$
H	$\rightarrow$	$H_1 \mid H_2 \mid \boldsymbol{(} H_3 \boldsymbol{)} \mid \boldsymbol{(} H_4 \boldsymbol{)}$
H_1	$\rightarrow$	$\boldsymbol{(} VWV \boldsymbol{)} \mid \boldsymbol{(} VWSL_1 \rrparenthesis \mid \boldsymbol{(} VWL_2 \rrparenthesis \mid \llparenthesis R_1SWV \boldsymbol{)} \mid \llparenthesis R_2WV \boldsymbol{)}$
		$\llparenthesis R_1SWSL_1 \rrparenthesis \mid \llparenthesis R_2WL_2 \rrparenthesis \mid \llparenthesis R_2WSL_1 \rrparenthesis \mid \llparenthesis R_1SWL_2 \rrparenthesis$
H_2	$\rightarrow$	$\boldsymbol{(} SL_1 \rrparenthesis \mid \boldsymbol{(} L_2 \rrparenthesis \mid \llparenthesis R_1S \boldsymbol{)} \mid \llparenthesis R_2 \boldsymbol{)} \mid \llparenthesis VWV \rrparenthesis \mid \llparenthesis R_1SL_1 \rrparenthesis \mid \llparenthesis R_1L_1 \rrparenthesis$
H_3	$\rightarrow$	$\boldsymbol{(} H_3 \boldsymbol{)} \mid H_1 \mid DHS \mid HS \mid DH \mid D$
H_4	$\rightarrow$	$\boldsymbol{(} H_4 \boldsymbol{)} \mid H_2$
V	$\rightarrow$	$\boldsymbol{(} V \boldsymbol{)} \mid H_1 \mid H_2 \mid DHS \mid HS \mid DH \mid D$
L_1	$\rightarrow$	$\boldsymbol{(} V \mid \llparenthesis R_2 \mid \llparenthesis R_1S \mid \llparenthesis R_1SWV \mid \llparenthesis R_2WV \mid \boldsymbol{(} VWV$
L_2	$\rightarrow$	$\boldsymbol{(} V \mid \llparenthesis R_2 \mid \llparenthesis R_1S \mid \llparenthesis R_1SWV \mid \llparenthesis R_2WV \mid \boldsymbol{(} VWV$
R_1	$\rightarrow$	$V \boldsymbol{)} \mid L_2 \rrparenthesis \mid SL_1 \rrparenthesis \mid VWSL_1 \rrparenthesis \mid VWL_2 \rrparenthesis \mid VWV \boldsymbol{)}$
R_2	$\rightarrow$	$V \boldsymbol{)} \mid L_2 \rrparenthesis \mid SL_1 \rrparenthesis \mid VWSL_1 \rrparenthesis \mid VWL_2 \rrparenthesis \mid VWV \boldsymbol{)}$
D	$\rightarrow$	$* \mid *D$
W	$\rightarrow$	$\boldsymbol{)(} VW \mid \boldsymbol{)(}$

Stems

To count the number and length of stems, we modify G to make the production of unpaired nucleotides explicit by introducing intermediate D with rule $D \to * \mid *D$ (see discussion of G_{ru}). However, instead of modifying the rules for S to incorporate D (as done for G_{ru}) we introduce another nonterminal V with $V \to D \mid DH \mid DHS \mid H \mid HS$. Note that according to above discussion, V and S are equivalent and thus can be interchanged arbitrarily without altering the generated language or introducing ambiguities. However, having a second means to generate a complete extended secondary structure will help with marking stems (at some places we will use S, at other V depending on whether we need a production to contribute to our marking or not). Now we take a look at all the conclusions of H in G. We aim at distinguishing 4 cases: First, productions that whenever applied introduce new novel stems (by initiating a new bond chain that cannot extend an existing stem due to ramifications). Second, productions that give only rise to a new novel stem if the pair of brackets generated is not directly nested in the last base pair of another stem, third, purely classic stems and forth those stems of classic base pairs which change into a novel stem by the production of a base triple. The following table shows all the right-hand sides of H-productions and the resulting variants that follow from

- translating each occurrence of T into two alternatives, one where $T \to \varepsilon$ is used, i.e., T is erased and one for $T \to S$, and
- replacing S by V if appropriate.

Furthermore, we specify, in which cases that productions imply a contribution to the number of novel stems.

Conclusion from G	Derived variants for G_s	Contribution to novel stem(s)
⸨$RTWTL$⸩	⸨$RSWSL$⸩, ⸨RWL⸩, ⸨$RWSL$⸩, ⸨$RSWL$⸩	always
⸨RTL⸩	⸨RL⸩, ⸨RSL⸩	only if not directly nested
⸨SWS⸩	⸨VWV⸩	only if not directly nested
(S)	(V)	eventually if base triple follows
(SWS)	(VWV)	always
⸨$RTWS$)	⸨RWV), ⸨$RSWV$)	always
($SWTL$⸩	(VWL⸩, ($VWSL$⸩	always
⸨RT)	⸨R), ⸨RS)	only if not directly nested
(TL⸩	(SL⸩, (L⸩	only if not directly nested

The same way, we modify the L, R and W productions, incorporating the rule used for T into their right-hand sides. Now we separate H into three nonterminals H_1 (always), H_2 (if not directly nested) and H_3 (eventually) together with the productions $H \to H_1 \mid H_2 \mid H_3$ in order to distinguish the three cases in above table for a former H-production to contribute a novel stem. This introduces only a single production for H_3, namely $H_3 \to$ (V). We thus can replace H_3 by (V) and change the set of productions for V into $V \to$ (V) $\mid D \mid DH \mid DHS \mid H_1 \mid H_2 \mid HS$, substituting H in $V \to H$ by the three possible conclusions for H. Now we are almost done. When aiming for a marking of the production rules that give rise to a new novel stem while setting up generating functions, we observe that we must distinguish two "kinds" of usages of nonterminal V, one where we have to mark the use of production $V \to H_2$ and one where we have not. Therefore we reuse H_3, copying the set of productions for V and replacing $H \to H_3$ by $H \to (H_3)$ which obviously is without effect to language and ambiguity. Furthermore, we observe that an L- resp. R-production might initiate a new novel stem (this is the case if neighbored by S resp. V in the sentential form, i.e., in the right-hand side of the H_i-rule producing L resp. R) or just might extended an existing stem. Thus we have to double both intermediates. Introducing symbols L_1, L_2, R_1 and R_2 assumed to be distinguished copies of L and R (i.e. assumed to have the same set of production) allows to take care of before observation while counting. To this end, we replace L (resp. R) in the right-hand sides of H_i-productions by L_1 (resp. R_1) if neighbored by S and by L_2 (resp. R_2) otherwise. The same is done for the conclusions of L_i resp. R_i rules themselves. This way, marking any application of an L_1 or R_1 rule by y (or by y^2) counts the corresponding newly initiated novel stems. Last but not least we need to distinguish between two kind of substrings emerging from (H_3): those which show up in a novel stem and those which are part of a classic stem (according to before discussion, both is possible for strings derived from H_3). Accordingly, we introduce another intermediate H_4 which takes care of such base pairs within novel stems and use H_3 only in connection with classic stems (note that the intermediates H_i now correspond to the 4 cases for H rules distinguished above). Thus production $H \rightarrow (H_3)$ becomes $H \rightarrow (H_3) \mid (H_4)$ and the rule set for H_3 is divided into $H_3 \rightarrow (H_3) \mid H_1 \mid DHS \mid HS \mid DH \mid D$ (a classic stem initiated by $H \rightarrow (H_3)$ is either extended, or terminated by a ramification or a bulge, or terminated by a hairpin loop) and $H_4 \rightarrow (H_4) \mid H_2$ (a stem initiated by $H \rightarrow (H_4)$ is extended but eventually becomes a novel stem by switching to nonterminal H_2 which produce at least one base triple belonging to the stem considered). At the end of the process we arrive at the grammar G_s shown in Table 12 which by construction generates the same language as G. By comparing the corresponding enumerator generating functions it is an easy exercise to conclude that no ambiguities have been introduced. Here it is possible to consider all stems just by adding up what is observed for classic and novel stems. However, for the number of bases in stems, in order to obtain feasible results for mean and proportion as well as for the expected length we counted the contribution of bases in two different ways. As we have observed above, a base triple can participate in two stems. However, counting its contribution only once (which e.g. is needed for the proportion of *Bases in novel stems of all bases*) is simple since the number of bases in all stems corresponds exactly to the sum of the number of pair bases and the number of base triples. Counting a symbol $\{(,),)(\}$ once or twice depending on the number of stems it participates becomes possible by marking the corresponding productions in G_s by y or y^2 respectively.

Bulges and interior loops

The process to derive a grammar for counting number and length of bulges is similar to what we did for stems. Starting from G, we first make the derivation of unpaired nucleotides explicit by introducing intermediate D and the productions $D \rightarrow * \mid *D$. Then we introduce nonterminal V with $V \rightarrow D \mid DH \mid DHS \mid H \mid HS$ noting that this way V becomes equivalent to S. Thus we can replace S by V whenever needed. We replace $H \rightarrow (RTL)$ by $H \rightarrow (RL) \mid (RSL)$ observing that for the latter production S – even if generating a run of $*$ – will not give rise to a bulge. All other occurrences of S in the right-hand sides of H, L, R, T and W productions are replaced by V. Now we have three different origins of a bulge: A symbol T in the right-hand side of a H, L or R rule produces a run of symbols $*$ (if this is the case we always have a bulge which can easily be checked by inspecting the corresponding rules), or a symbol V in the right-hand side of a H, L, R or W rule producing a nested stem together with a run of symbols $*$ to its left or right. To identify the two last cases, we introduce a copy of symbol V, using names V_1 and V_2 to denote the two different replica; V_1 and V_2 have – so far – the same rules (those of former V). We replace V for $T \rightarrow V$ by V_1,

Table 13: Grammar G_b used for counting the number and total length of bulges.

S	$\rightarrow$	$\ast \mid \ast S \mid H \mid HS$
H	$\rightarrow$	$⸨RTWTL⸩ \mid ⸨RL⸩ \mid ⸨RSL⸩ \mid ⸨V_2WV_2⸩ \mid (V_2) \mid (V_2WV_2) \mid ⸨RTWV_2) \mid (V_2WTL⸩ \mid ⸨RT) \mid (TL⸩$
L	$\rightarrow$	$⸨RTWV_2 \mid (V_2WV_2 \mid (V_2 \mid ⸨RT$
R	$\rightarrow$	$V_2WTL⸩ \mid V_2WV_2) \mid V_2) \mid TL⸩$
T	$\rightarrow$	$\epsilon \mid V_1$
W	$\rightarrow$	$)(V_2W \mid)($
V_1	$\rightarrow$	$D_2 \mid D_1H \mid D_1HS \mid H \mid HS$
V_2	$\rightarrow$	$D_1 \mid D_2H \mid D_1HS \mid H \mid HH \mid HD_2 \mid HHS \mid HD_1H \mid HD_1HS$
D_1	$\rightarrow$	$\ast \mid \ast D_1$
D_2	$\rightarrow$	$\ast \mid \ast D_2$

Table 14: Grammar G_{il} used for counting the number and total length of interior loops.

S	$\rightarrow$	$\ast \mid \ast S \mid H \mid HS$
H	$\rightarrow$	$⸨RTWTL⸩ \mid ⸨RTL⸩ \mid ⸨VWV⸩ \mid (V) \mid (VWV) \mid ⸨RTWV) \mid (VWTL⸩ \mid ⸨RT) \mid (TL⸩$
L	$\rightarrow$	$⸨RTWV \mid (VWV \mid (V \mid ⸨RT$
R	$\rightarrow$	$VWTL⸩ \mid VWV) \mid V) \mid TL⸩$
T	$\rightarrow$	$\epsilon \mid S$
W	$\rightarrow$	$)(VW \mid)($
V	$\rightarrow$	$H \mid HS \mid D_1 \mid D_1H \mid D_2HD_2 \mid D_1HH \mid D_1HHS \mid D_1HD_1H \mid D_1HD_1HS$
D_1	$\rightarrow$	$\ast \mid \ast D_1$
D_2	$\rightarrow$	$\ast \mid \ast D_2$

counting a bulge whenever V_1 generates a run of symbols $\ast$ only. At all other places V is replaced by V_2. However, we still cannot uniquely identify the generation of a bulge according to the two other possibilities, since $V_2 \rightarrow HS$ might either contribute a bulge ($S \Rightarrow^\star \ast^+$) or not ($S$ produces something different from $\ast^+$). Therefore we expand the productions of S in $V_2 \rightarrow HS$ one step further. To this end we replace S within the conclusion by V_1 and then V_1 by the conclusions of all V_1 productions, getting $V_2 \rightarrow HH \mid HD_2 \mid HHS \mid HDH \mid HDHS$ together with the other V_2 productions which have not been changed. Finally, we introduce a copy of D – using intermediates D_1, D_2 – using D_2 for D whenever a bulge is generated, D_1 otherwise. In detail, for $V_1 \rightarrow D$, $V_2 \rightarrow DH$ and $V_2 \rightarrow HD$ we replace D by D_2 (those are the three cases giving rise to a new bulge distinguished above), all the other occurrences are replaced by D_1. This way we arrive at grammar G_b shown in Table 13. By before discussion it should be obvious that the language generated has not been altered. To see that no ambiguities have been introduced we compare G's and G_b's counting generating functions.

Grammar G_{il} used to consider interior loops is constructed in an analogous way. The production of symbols $\ast$ is made explicit introducing D and V together with $D \rightarrow \ast \mid \ast D$ and $V \rightarrow D \mid DH \mid DHS \mid H \mid HS$ noting again that this way V becomes equivalent to S. Afterwards, S is replaced by V in the right-hand sides of any H, L, R and W production. To uniquely identify interior loops, $V \rightarrow DHS$ must be extended one step further by replacing S by any possible V conclusion. This way we arrive at $V \rightarrow H \mid HS \mid D \mid DH \mid DHD \mid DHH \mid DHHS \mid DHDH \mid DHDHS$. Now an interior loop is generated if and only if $V \rightarrow DHD$ is applied. To be able to count the length of the loop generated we introduce a copy of D, using D_2 for $V \to D_2 H D_2$ and D_1 of any other occurrence of D. Table 14 shows the obtained grammar G_{il} at full detail. For the same reasons as stressed before, G_{il} can be used for counting number and length of interior loops.