Classification and predictions of RNA pseudoknots based on topological invariants

Graziano Vernizzi, Henri Orland, and A. Zee

Phys. Rev. E **94**, 042410 — Published 12 October 2016

DOI: 10.1103/PhysRevE.94.042410

A new topological invariant for the classification and prediction of RNA pseudoknots

Graziano Vernizzi[1] , Henri Orland[2,3,4] and A. Zee[2,4,5]
[1] Department of Physics and Astronomy, Siena College, New York, USA
[2]Institut de Physique Théorique, CEA Saclay, 91191 Gif-sur-Yvette Cedex, France
[3]Beijing Computational Science Research Center, Haidian District Beijing, 100084, China
[4]Department of Physics, University of California, Santa Barbara, CA 93106, USA
[5]Kavli Institute for Theoretical Physics, University of California, Santa Barbara, CA 93106, USA

September 13, 2016

Abstract

We propose a new topological characterization of RNA secondary structures with pseudoknots based on two topological invariants. Starting from the classic arc-representation of RNA secondary structures, we consider a model that couples both I) the topological genus of the graph and II) the number of crossing arcs of the corresponding primitive graph. We add a term proportional to these topological invariants to the standard free energy of the RNA molecule, thus obtaining a novel free energy parametrization which takes into account the abundance of topologies of RNA pseudoknots observed in RNA databases.
Keywords: Secondary structure, pseudoknot, RNA, structure classification, topology.
PACS: 82.39.Pj, 87.14.Gg

1 Introduction

The prediction of possible foldings of RNA molecules is still a major open problem of molecular biology[1;2]. It is of utmost importance, since the three-dimensional structure of any folded biopolymer mostly determines its biological function by providing the adequate geometry for biochemical reactions to occur. In the last thirty years, the role of RNA has been upgraded from being a relatively minor player in the central dogma of Watson and Crick to being one of the central players in molecular biology[1]. It has been recognized that in addition to being a carrier of genetic information, some RNA may also have enzymatic roles, and may play a central part in the regulation of biological networks[1]. In spite of considerable effort, the accurate prediction of the three-dimensional structure of RNA from its primary sequence has resisted so far the most advanced computational methods, in particular for long RNA sequences. In such cases, drastic approximations are necessary. A typical simplifying assumption is that the RNA secondary structure (i.e. the complete list of paired nucleotides) already provides sufficient information on the active sites of the RNA molecules, by allowing the identification of loops and other motifs such as pseudoknots, where the biochemistry takes place[3]. The energetic landscape of an RNA molecule is mostly dominated by Crick-Watson base pairings (A,U), (G,C), and the additional wobble pair (G,U). Non-canonical base pairs and tertiary interactions have been recognized to further stabilize the structure[4] of RNA, nonetheless we will not include them in the present work. Several deterministic and stochastic methods have been proposed for the prediction of secondary structures of RNA molecules[5;6;7;8]. Despite great progress, their overall success is limited, in particular for long RNA molecules. Part of the difficulty lies in the prediction of RNA pseu-

Email addresses: gvernizzi@siena.edu (Graziano Vernizzi), Henri.Orland@cea.fr (Henri Orland), zee@kitp.ucsb.edu (A. Zee).
*All authors contributed equally to this work.

doknots, which has been identified as an NP-complete problem[9].

2 Statistical physics of RNA pseudoknots

We now summarize some standard notations to represent all base-pairings in a RNA molecule. The backbone of an RNA molecule can be represented by an oriented straight line (from the $5'$ to the $3'$ end), on which the nucleotides appear in the order given by the RNA primary sequence. A pairing between two bases is depicted by an arc joining the two bases in the upper half-plane above the backbone line (see Fig. 1). A graph without crossing pairing lines is called a *planar graph*. If a graph contains lines that cross, then it is said to contain a *pseudoknot*. If one assigns a suitable

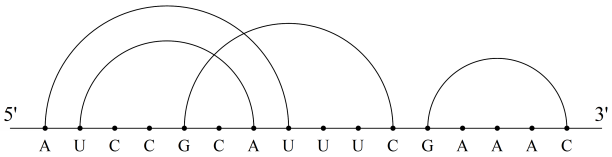

Figure 1: In the arc-diagram representation of an RNA, crossing arcs indicate the presence of a pseudoknot.

pairing energy (called *stacking energy*) to adjacent base pairs, then it is possible to compute the partition function of all planar graphs exactly, by using standard recursion equations[10;11]. However, if one allows for the occurrence of pseudoknots then those recursive algorithms face an overwhelming increase in polynomial complexity. Several alternative algorithms have been proposed to predict pseudoknotted structures[12;13;14;15;16].

We have proposed a topological classification of pseudoknots in terms of their genus[15], followed by two algorithms for the prediction of such pseudoknots[16;17]. The genus of an RNA graph can be defined in the following way[18]: join the $5'$-end with the $3'$-end by bending the backbone line in the lower-half plane to make a circle, so that all pairing lines exist on the outside of the circle. The actual size of such a circle is of course irrelevant, and it is therefore topologically equivalent to a puncture on a surface. The genus of the graph is the minimal number of handles one has to carve in a punctured sphere, so that the graph can be drawn on it without any crossing. A planar graph by definition can be drawn on a sphere without any crossing arc, and so it is of genus $g = 0$ (the sphere has no handles). A H-pseudoknot (i.e. the "ABAB" pseudoknot with two helices A and B) can be drawn without crossing on a torus, which is a sphere with one handle and therefore with genus $g = 1$ (see Fig. 2).

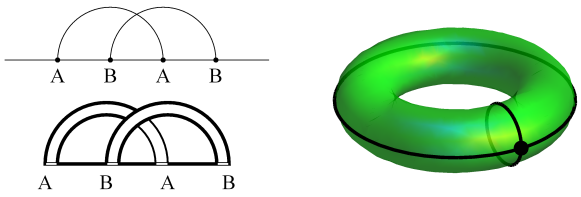

Figure 2: The arc-diagam representation of an "ABAB" H-pseudoknot, its double-line representation, and its embedding on a torus on which the arc-diagram can be drawn without crossings. This corresponds to a topological genus $g = 1$.

A practical diagrammatic way to compute the genus of a graph is by using the so-called double-line representation, where base-pairs are drawn using oriented double lines (see Fig. 2). In such a representation, oriented loops appear on the graph. The genus can be shown to be equal to $g = (p - l)/2$ where p is the number of pairings of the graph (i.e. the number of arcs) and l is the number of closed loops. The genus allows to organize pseudoknots and secondary structures of RNA systematically in equivalence classes, each class corresponding to a value of the genus g[19;15]. It is a topological invariant which depends only on the connectivity of the RNA base-pairs. Moreover, it has the property of being additive: if a structure comprises two consecutive pseudoknots with genus g_1 and g_2, the genus of the whole RNA sequence is $g = g_1 + g_2$. However, it is known experimentally that pseudoknots are fairly rare in RNA molecules[19]. Furthermore, they usually impose some mechanical constraint on the sugar-

phosphate backbone of the molecule. We have thus proposed[16;17] to add an energetic penalty proportional to the genus, to the standard folding energy (which includes stacking energies, loop penalties, etc.). Within such a framework, the partition function of the system is

$$\mathcal{Z} = \sum_{all\ graphs} e^{-\beta[E(graph)+\mu_g g(graph)]} \qquad (1)$$

where $\beta = 1/k_B T$ is the inverse temperature, k_B is the Boltzmann constant, E is the free energy (which phenomenologically includes the configurational entropy at fixed genus) and g is the topological genus. The parameter μ_g is a phenomenological parameter, used to penalize graphs with high genus. Planar graphs, i.e. graphs without pseudoknots are obtained by taking μ_g to infinity[18;20].

We have developed two algorithms to sample the partition function in eq. (1) and predict the secondary structures of RNAs with pseudoknots[16;17]. In[16], we first make a library of possible paired RNA segments from the sequence. We then enumerate all the possible assemblies of these fragments and compute the corresponding free energy. The minimal free energy state can be computed, but the method is limited to fairly small sizes ($L < 150$ where L is the number of nucleobases). In[17], we start from the same library of building blocks, but we assemble them using a Monte Carlo algorithm (multiple Markov chains). This last method allows to handle RNAs of sizes up to 1000 nucleobases.

Although methods based on eq. (1) are promising, they do not predict correctly the abundance of various structures with identical genus and comparable energy. That is mostly due to the fact that typical energy functions do not explicitly discriminate between structures with identical genus. For example, following ref.[19], there are four primitive graphs of genus $g = 1$. We define a *primitive graph* as a graph which is both irreducible (i.e. cannot be disconnected by cutting the backbone somewhere) and non-nested (i.e. cannot be disconnected by cutting *twice* the backbone somewhere), and in which all equivalent parallel pairing arcs (i.e. a *sheaf* of parallel arcs) are collapsed into a single *renormalized* arc. Later in this paper, we give an alternative definition, but completely equivalent. In Fig. 3, we sketch all four primitive graphs with genus $g = 1$ (which have been obtained first in ref.[21] by steepest descent methods).

With obvious notations, the 4 pseudoknots can be labeled as ABAB, ABACBC, ABCABC, ABCADBCD. As it was shown in ref.[19], the abundance of ABAB, either in the databases PDB or in PseudoBase, is much larger than that of ABACBC. The ABCABC pseudoknot is quite rare while the ABCADBCD is absent from the databases. This variation in abundance of the various genus 1 primitive pseudoknots is hardly understandable if the energetic penalty is only dependent on the genus. In fact, while it would be straightforward to create a sequence that has minimum energy for the ABACBC configuration, by simply inverting the inner "AC" into "CA" it can be rendered a ABCABC pseudoknot with similar energy and similar genus, but which is far more rare in nature. To account for such a variation within

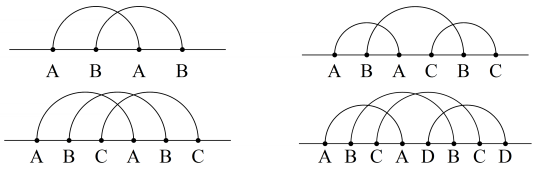

Figure 3: There are four type of primitive pseudoknots with genus $g = 1$.

a given genus, it is thus necessary to supplement the free energy by an additional term which would further discriminate between the structures. If we look at the four graphs of fig. 3, we see that they differ by the number of crossings (i.e. crossing arcs) of the effective pairing arches. The ABAB graph has 1 crossing, ABACBC has 2 crossings, ABCABC has 3 crossings and ABCADBCD has 5 crossings. It turns out that their abundance decreases as a function of the number of crossings.

3 Chemical potential for RNA crossing arcs

As the number of crossings of a primitive graph is an additive quantity, it is natural to include an energetic penalty proportional to this number.

We therefore introduce the *renormalized crossing number* in the following way:

1. Given a generic graph $\mathcal{D}$, let $\mathcal{D} = \mathcal{D}_1 + \mathcal{D}_2 + \ldots$ be its decomposition in irreducible or nested parts $\mathcal{D}_i$.

2. For each graph $\mathcal{D}_i$ we consider its primitive version $\mathcal{D}_i'$ (i.e. all stacked arcs are collapsed into a single renormalized arc).

3. The renormalized crossing number N_c of $\mathcal{D}$ is defined as the sum of the crossing number of each $\mathcal{D}_i'$.

Such a definition allows to generalize the free energy for a RNA graph:

$$E_1 = E(graph) + \mu_g g(graph) + \mu_c N_c(graph)\,, \quad (2)$$

and

$$\mathcal{Z} = \sum_{all\ graphs} e^{-\beta E_1}\,, \quad (3)$$

where $N_c(graph)$ denotes the renormalized crossing number of a given *graph*, and μ_c controls the associated energetic penalty. As was shown in ref.[16], a typical value for the genus penalty is $\beta\mu_g = 1.5$.

The crossing penalty can be estimated by trying to fit the abundance of the various types of genus 1 pseudoknots. Currently, there are 398 pseudoknots in the *Pseudobase* database[22]. In particular there are 355 ABAB graphs, 7 ABACBC graphs, 1 ABCABC graph, and no ABCADBCD graphs (for a total of 363 pseudoknots with genus 1). Such an "exponential" decay can be roughly described by using an approximate value of $\beta\mu_c \approx 2.5$. Such an estimate represents an empirical average over several RNA sequences, and therefore it neglects individual RNA sequence biases within a given isogenus population. This provides a convenient way to account for the under-represented abundance of the ABCABC pseudoknot and the absence of the ABCADBCD pseudoknot, both of genus 1. It is worth emphasizing here that the relative abundance of different pseudoknot classes, can be described by introducing a *single* linear term in the free energy, with a the corresponding phenomenological parameter μ_c. Furthermore, in Pseudobase there are also 35 ABCDCADB graphs with genus 2. The latter represent a slightly biased sample since they *all* are of the HDV-like ribozyme type (see the diagram on the second column, third row of Figure 4). A more systematic fit of the genus and number crossing penalties will be presented in a forthcoming study.

An important remark is in order at this point: the genus and the crossing number do not uniquely specify an RNA graph. Indeed, it is easy to see that except for $g = 1$ there may exist several graphs with same genus and crossing number. In Figure 4 we display 8 graphs with genus $g = 2$ and crossing number $N_c = 3$. Note that the introduction of a crossing penalty μ_c requires to recompute the value of the genus penalty μ_g. A more precise determination of both penalties will be performed in a forthcoming study, by optimizing them in order to improve the success rate of the prediction algorithms.

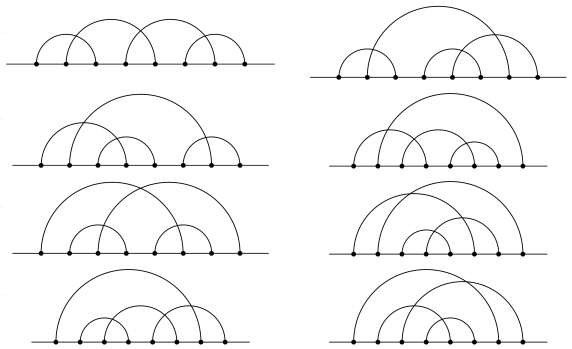

Figure 4: There are 8 primitive pseudoknots with genus $g = 2$ and crossing number $N_c = 3$.

We conclude this Section with a remark on the link between the crossing number and the topological genus. References[24] and[25] provide an upper bound and lower bound, respectively, for the crossing number of any graph with L vertices, maximum degree d, and genus g:

$$\alpha dLg \leq N_c \leq dLc^g\,, \quad (4)$$

In our approach,the effect due to the length of the helices is included only via the energy function, and it is not introduced explicitly here. For instance, the asymmetries in the stem and loop lengths for most ABAB-pseudoknots are well explained by a single-parameter thermodynamic model presented in[23].

where c and α are constants, with $c > 1$ and $\alpha > 0$. In the RNA case one has $d = 3$. Moreover, in[26] we showed that an RNA molecule with genus g need to contain at least $L = 4g$ nucleobases, i.e. $L/4 \geq g$. Therefore, one has:

$$12\alpha g^2 \leq N_c \leq 3Lc^{L/4} \,. \tag{5}$$

The upper bound in eq. (5) shows that large N_c values require a large L, i.e. long RNA primitive diagrams. On the other hand, the lower bound shows that small N_c values (i.e. fewer crossings) are inevitably linked with small values of g. As suggested in[19], a possible reason for the empirical bias towards pseudoknots with low genus is that evolutionary pressures have led to complex pseudoknots built from many small primitive pseudoknots with low genii (i.e. longer structures are simply built out of simpler structures). It is in this spirit that we introduced a chemical potential μ_c for the crossing number (which is an additive quantity) to the free energy eq. (2).

4 Implementation and Algorithms

In this section we describe some algorithms to a) extract the primitive graph from any RNA diagram, b) to compute its genus and c) its renormalized crossing number. For practical software implementations it is convenient to represent the pairing of a generic RNA diagram by using a formalism based on permutations. Given an RNA sequence with L bases, each base can be identified by an integer number $i = 1, \ldots, L$, from the $5'$ end to the $3'$ end. A specific pairing (i, j) is denoted by a permutation π with $\pi(i) = j$. Obviously, pairings are symmetric and therefore also $\pi(j) = i$ holds true. Unpaired bases are represented by "fixed points" $\pi(i) = i$. With such conventions, the permutation is an *involution*, that is $\pi(\pi(i)) = i$ for all $i = 1, \ldots, L$.

4.1 Irreducible diagrams

To decompose any RNA graph in its irreducible components, is sufficient to verify recursively whether it can be disconnected by cutting the backbone at any one point (see fig. 5). In particular, we may use an electrostatic analogy where the pairings and the backbone are regarded as electrostatic field lines. We assign a positive charge $q = +1$ to every base where a pairing begins, (i.e. with $\pi(i) > i$), a negative charge $q = -1$ to every base where a pairing ends, (i.e. with $\pi(i) < i$), and no charge, $q = 0$, to every free base (i.e. with $\pi(i) = i$):

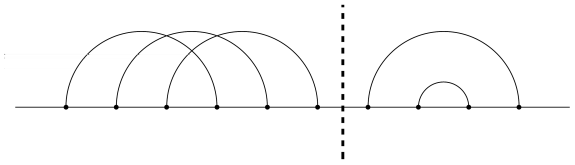

Figure 5: A reducible diagram can be decomposed in two disconnected components by cutting the backbone once (dotted line).

$$q_i = sign\left(\pi(i) - i\right) \,, \quad i = 1, \ldots, L \,, \tag{6}$$

where *sign* is the sign function (equal to 0 for vanishing argument). When the cumulative sum $c_k = \sum_{i=1}^{k} q_i$ is zero, then all pairings that started before the k-th base also must have ended before the k-th base. In fact, the RNA segment up to the base k (included) is charge neutral, and thus is loosely bound to the rest of the molecule (i.e. there are no unbalanced pairings to the left of k). By cutting the backbone just on the right of the k-th base, the molecule disconnects into two separate components. This procedure can be repeated all the way to the 3' end of the RNA molecule (up to $i = L$), and every time that the cumulative sum c_k is zero, the graph can be disconnected by cutting the backbone at the base k. The pseudocode implementing such procedure is in Algorithm 1.

4.2 Nested diagrams

The next essential tool is the identification of all nested components in the diagram. A diagram is said to contain a nested component if such a component can be removed by cutting the backbone at *two* points. The concept of "nestedness" is closely related to the concept of irreducibility. This can be illustrated by introducing the cyclic (right) shift-permutation $\sigma = (2, 3, 4, \ldots, L, 1)$. Under

Algorithm 1 Decompose a diagram into irreducible components

Require: π (a permutation involution)

```
L=length(π)
StartsAt=1;c=0
for i=1 to L do
   c=c+sign(π(i)-i)
   if c is 0 then
      Print "irreducible from " StartsAt "to" i
      StartsAt=i+1;
   end if
end for
```

such shift permutation, every site i is mapped onto its right-neighbor $i+1$. Moreover, the permutation is cyclic in the sense that the last base $i = L$ is mapped onto the first one $i = 1$. By applying the shift permutation σ a sufficient number of times, any nested component of the diagram can be translated to the right until its rightmost base touches $i = L$. Such a diagram is reducible evidently. Therefore, one can identify all nested components of a diagram by simply identifying all the irreducible parts of $\sigma^k \cdot \pi$ for all $k = 1, \ldots, L$. Such a procedure is implemented in Algorithm 2.

Algorithm 2 To identify all irreducible and nested components

Require: π (a permutation involution)

```
L=length(π)
σ = (2,3...,L,1) (the cyclic shift permutation)
for i=1 to L do
   find (and output) all irreducible components of π (use Algorithm 1)
   π = σ · π;
end for
```

We note that all free bases are by definition also nested components, since it is possible to disconnect any free base i by simply cutting the backbone at $i-1$ and $i+1$. Therefore, when considering diagrams that do not have any nested component, one can as well consider diagrams where all free bases are removed.

The possibility of identifying all nested components in a RNA diagram, opens the way to a procedure that we defined as "backbone renormalization" in [18]. It consists of replacing each nested component by a new type of backbone segment, called z_g, where g is the genus of the nested component that has been replaced. To that objective, we briefly review [18] how to compute the genus of any diagram (nested or not, irreducible or not).

4.3 The genus

The explicit evaluation of the formula $g = (p-l)/2$ can be performed efficiently by using the formalism of permutations. In this case, the number of pairings, which is simply half the number of paired bases, is given by

$$p = \frac{1}{2}\sum_{i=1}^{L}(1 - \delta_{i\pi(i)}) \tag{7}$$

where δ is the Kronecker delta function. The total number of loops can be obtained by counting the number of cycles c of the permutation $\sigma \cdot \pi$ where σ is the cyclic shift-permutation [27]. One can easily verify that $c = l+1$, that is, among all cycles there is also a loop which contains the cyclic link from $i = L$ to $i = 1$. We have:

$$g = \frac{p - c + 1}{2}\,. \tag{8}$$

The pseudocode to compute the genus of a permutation involution π is in Algorithm 3.

Algorithm 3 Compute the genus of the diagram π

Require: π (a permutation involution)

```
L=length(π)
σ = (2,3...,L,1) (the cyclic shift permutation)
τ = σ · π
c = number of cycles of τ
p = (L− number of fixed points of(π) )/2;
genus=(p − c + 1)/2
```

It is straightforward to verify also that the genus is an additive quantity both in the nested components and in the irreducible parts. More precisely, if the (reducible) diagram $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2$ is the sum of two irreducible components $\mathcal{D}_1, \mathcal{D}_2$, then

$g(\mathcal{D}) = g(\mathcal{D}_1) + g(\mathcal{D}_2)$. Analogously, if the diagram $\mathcal{D}$ has a nested component $\mathcal{D}_1$, then again $g(\mathcal{D}) = g(\mathcal{D}_1) + g(\mathcal{D}_2)$, where $\mathcal{D}_2$ is the complement of $\mathcal{D}_1$ in $\mathcal{D}$.

4.4 Primitive diagrams

The final requirement to characterize primitive diagrams is to collapse parallel pairing lines in the graph into a single one. We say that two lines (or arcs) are *equivalent* if they don't cross, and if they intersect exactly the same pairing lines (see Fig. 6). A simple way to translate it into an algorithm is to define a primitive diagram as an irreducible, not nested diagram, with no stacked pairings. Any stacked pairing in π corresponds to a cycle of length two for the composite permutation $\bar{\sigma}\cdot\pi$, where $\bar{\sigma} = \{2, 3, 4, \ldots, L, L\}$ is the *non-cyclic* (right) shift permutation. Therefore, a primitive diagram is represented by a permutation involution π which is irreducible, not nested and such that $\bar{\sigma}\cdot\pi$ does not contain any cycle of length two. A simple algorithm to "renormalize" nested arcs in a generic diagram, and to adsorb any nested planar diagram into renormalized backbones (of planar type only) is listed in Algorithm 4.

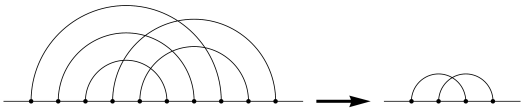

Figure 6: Arc renormalization: any sheaf of parallel arcs can be mapped into a single arc

Algorithm 4 Primitive diagram

Require: π (a permutation involution)

flag=1
while flag=1 **do**
  flag=0; L=length(π)
  $\sigma = (2, 3 \ldots, L, L)$ (not-cyclic shift permutation)
  for i=1 to L **do**
    if $\sigma(\pi(\sigma(\pi(i)))) = i$ **then** {if there is a 2-cycle, then replace one of the two arcs with free bases.}
      $\pi(\pi(i)) = \pi(i)$; $\pi(i) = i$
    end if
  end for
  counter=1 {Relabel the sequence, while skipping all fixed points}
  for j=1 to L **do**
    if $\pi(j) = j$ **then**
      label(j)=0
    else
      label(j)=counter; counter++
    end if
  end for
  for j=1 to L **do**
    if $\pi(j)$ is not equal to j **then**
      π_{new}(label(j))=label(π(j))
      flag=1
    end if
  end for
  $\pi = \pi_{new}$
end while

4.5 The renormalized crossing number

We conclude this section by providing an algorithm to compute the crossing number and the renormalized crossing number of a generic diagram. The crossing number is the lowest number of crossing points among arcs in the diagram. Like the genus, the crossing number is also an additive quantity with respect to nestedness and reducibility. In simple words, the crossing number of any reducible (or nested) diagram $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2$ is $N_c(\mathcal{D}) = N_c(\mathcal{D}_1) + N_c(\mathcal{D}_2)$. However, as we have discussed previously the crossing number is not invariant under arc-renormalization: for instance,

the crossing number of the diagrams in Fig. 6 is $N_C = 6$ for the graph on the left and $N_C = 1$ for graph on the right. Algorithm 5 parses the RNA permutation involution π and for each arc $(\pi(i) > i)$, first it counts the number of intersecting arcs between i and $\pi(i)$, and then removes it. While Algorithm 5 works for any graph, including primitive ones, in our thermodynamic model eq. (3) only the renormalized crossing number of a graph is necessary. As explained in the introduction, the rationale is that RNA databases do not show a preference for short vs. long helices for same-genus pseudoknots, in addition to the enthalpic contribution. Algorithm 6 outlines the pseudocode for computing the renormalized crossing number.

Algorithm 5 Compute the crossing number of π

Require: π (a permutation involution)

```
L=length(π)
CrossingNumber=0
for i = 1 to L do
  if π(i) > i then {rising arc}
    a=i
    b=π(i)
    for j = a + 1 to b − 1 do {for each base inside the arc}
      if π(j)>b then
        CrossingNumber++
      end if
    end for
    π(π(i)) = i
    π(i) = i
  end if
end for
```

By using these algorithms to compute the genus and the crossing number of a graph, it is possible to perform a Monte Carlo sampling of graphs of the system analogous to ref.[17], using the energy of eq. (3). The full implementation of the algorithm and the fitting of the genus and crossing number penalties will require additional work which will be presented in a forthcoming paper.

Algorithm 6 Compute the renormalized crossing number N_c of π

Require: π (a permutation involution)

```
Use Algorithm A2 to find all m irreducible and nested components D_i of π.
N_c=0
for i = 1 to m do
  Use Algorithm A4 to compute the primitive diagram D'_i of D_i.
  Use Algorithm A5 to compute the crossing number n_c of D'_i.
  N_c = N_c + c
end for
```

5 Conclusions

In addition to a topological chemical potential coupled to the genus, we propose to add a term proportional to the renormalized crossing number of a RNA graph to the energy function of pseudoknotted RNAs. Such a procedure requires the systematic evaluation of the primitive diagram of any RNA secondary structure, with or without pseudoknots. In turn, that can be expressed naturally with the formalism used in matrix quantum field theory to renormalize Feynman diagrams. We discussed two levels of renormalization: the backbone and the arc renormalization, leaving the vertex renormalization to a future paper. The latter is helpful not only to collapse the RNA diagram into simpler ones, but can be used for building new diagrams with higher topological complexity from simpler ones. We are currently implementing the Monte Carlo algorithm with the modified energy function to predict RNA structures. In order to do so, it is necessary compute the change in the genus and in the renormalized crossing number of a graph upon addition or removal of a helical fragment (equivalent to a single pairing in the primitive graph). The incremental change of the genus was described in ref.[16], and the change of crossing number will be discussed in a forthcoming paper. However, as already pointed out in ref.[17], all these algorithms based on topology do not take into account the geometry of the molecule, and in particular, many of its predictions are plagued by steric clashes. The next challenge will be to include the

steric constraints at each step of the Monte Carlo procedure.

Acknowledgments: H.O. would like to thank Joel Hass for illuminating discussions, and the Physics Department of UCSB for its generous hospitality during part of this work. A.Z. is grateful to the CFM foundation for a chair of the ENS foundation, and to the NSF for grant PHY13-16748.

References

[1] D. Elliot and M. Ladomery, *Molecular Biology of RNA* (Oxford University Press, 2011).

[2] I. Tinoco Jr. and C. Bustamante, J. Mol. Biol. **293**, 271 (1999).

[3] D.W. Staple and S.E. Butcher, PLOS Biology, DOI: 10.1371/journal.pbio.0030213.

[4] N.B. Leontis, J. Stombaugh, and E. Westhof, Nucleic Acids Res. **30**,:3497-531 (2002).

[5] R. Nussinov, G. Pieczenik, J.R. Griggs, and D.J. Kleitman, SIAM Journal on Applied Mathematics, **35**(1):68-82, (1978).

[6] M. Zuker and P. Stiegler, Nucleic Acids Research **9** (1):133-148 (1981).

[7] D. Metzler and M.E. Nebel Journal of Mathematical Biology **56**(1):161-181 (2008).

[8] S. Bellaousov and D.H. Mathews, RNA **16**(10):1870-1880 (2010).

[9] R.B. Lyngso and C.N.S. Pedersen, Journal of Computational Biology **7**(3-4):409-427 (2000).

[10] M. Zuker, Nucleic Acids Research **31**(13):3406 (2003).

[11] S. Wuchty, W. Fontana, I. Hofacker, and P. Schuster, Biopolymers **49**, 145-165 (1999).

[12] J. Ren, B. Rastegari, A. Condon, and H.H. Hoos, RNA **11** (10):1494-1504 (2005).

[13] E. Rivas and S.R. Eddy Journal of Molecular Biology **285**:2053-2068, (1999).

[14] C.M. Reidys, F.W.D. Huang, J.E. Andersen, R.C. Penner, P.F. Stadler, and M.E. Nebel, Bioinformatics **27**(8):1076, (2011).

[15] H. Orland and A. Zee, Nucl. Phys. B **620**, 456-476 (2002).

[16] M. Bon and H. Orland, Nucl. Acids Res. doi: 10.1093/nar/gkr240 (2011).

[17] M. Bon, C. Micheletti and H. Orland, Nucl. Acids Res. **41** (3): 1895-1900 (2013).

[18] G. Vernizzi and H. Orland, *The Oxford Handbook of Random Matrix Theory*, chapter 42 (Oxford University Press, 2011).

[19] M. Bon, G. Vernizzi, H. Orland, and A. Zee, J. Mol. Biol. **379**, 900-911 (2008).

[20] M.G. dell'Erba and G.R. Zemba, Phys. Rev. E 80 041926 (2009).

[21] M. Pillsbury, H. Orland, and A. Zee, Phys. Rev. E **72**, 011911 (2005).

[22] F.H.D. van Batenburg, A.P. Gultyaev, C.W.A. Pleij, J. Ng,and J. Oliehoek, Nucl. Acids Res. **28**,1, 201-204 (2000).

[23] D. P. Aalberts and N.O. Hodas, Nucleic Acids Res. **33**, 2210-2214 (2005).

[24] J. Pach, G. Tóth, 13th Intl. Symposium on Graph Drawing, Lecture Notes in Computer Science 3843, Springer, Berlin, 334-342 (2006).

[25] H.N. Djidjev and I. Vrt'o, Automata, Languages and Programming, Volume 4051 of the series Lecture Notes in Computer Science, 419-430, Springer Berlin Heidelberg (2006).

[26] G. Vernizzi, H. Orland, A. Zee, Phys. Rev. Lett. **94**, 168103 (2005).

[27] J. Bouttier, *The Oxford Handbook of Random Matrix Theory*, chapter 26, (Oxford University Press, 2011).