



HAL
open science

Automatic exploration of the natural variability of RNA non-canonical geometric patterns with a parameterized sampling technique

Théo Boury, Yann Ponty, Vladimir Reinharz

► **To cite this version:**

Théo Boury, Yann Ponty, Vladimir Reinharz. Automatic exploration of the natural variability of RNA non-canonical geometric patterns with a parameterized sampling technique. WABI 2023 - 23rd Workshop on Algorithms in Bioinformatics, Texas A&M University, Sep 2023, Houston, United States. 10.4230/LIPIcs.WABI.2023.20 . hal-04094288v2

HAL Id: hal-04094288

<https://hal.science/hal-04094288v2>

Submitted on 24 Aug 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Automatic exploration of the natural variability of RNA non-canonical geometric patterns with a parameterized sampling technique

Théo Boury

Computer Science Department, Ecole Normale Supérieure de Lyon, France

Yann Ponty

Laboratoire d'Informatique de l'Ecole Polytechnique (CNRS/LIX; UMR 7161), Institut Polytechnique de Paris, France

Vladimir Reinharz

Department of Computer Science, Université du Québec à Montréal, Canada

Abstract

Motivation: Recurrent substructures in RNA, known as 3D motifs, consist of networks of base pair interactions and are critical to understanding the relationship between structure and function. Their structure is naturally expressed as a graph which has led to many graph-based algorithms to automatically catalog identical motifs found in 3D structures. Yet, due to the complexity of the problem, state-of-the-art methods are often optimized to find exact matches, limiting the search to a subset of potential solutions, or do not allow explicit control over the desired variability.

Results: We developed FuzzTree, a method able to efficiently sample approximate instances of an RNA motif, abstracted as a subgraph within a target RNA structure. It is the first method that allows explicit control over (1) the admissible geometric variability in the interactions; (2) the number of missing edges; and (3) the introduction of discontinuities in the backbone given close distances in the 3D structure. Our tool relies on a multidimensional Boltzmann sampling, having complexity parameterized by the treewidth of the requested motif. We applied our method to the well-known internal loop Kink-Turn motif, which can be divided into 12 subgroups. Given only the graph representing the main Kink-Turn subgroup, FuzzTree retrieved over 3/4 of all kink-turns. We also highlighted two occurrences of new sampled patterns. Our tool is available as free software and can be customized for different parameters and types of graphs.

2012 ACM Subject Classification Applied computing → Molecular structural biology

Keywords and phrases Subgraph Isomorphism, 3D RNA, Parameterized Complexity, Tree Decomposition, Boltzmann sampling, Neighborhood metrics, Kink-Turn family

Digital Object Identifier [10.4230/LIPIcs.WABI.2023.12](https://doi.org/10.4230/LIPIcs.WABI.2023.12)

Related Version A full version of the paper is hosted on HAL.

Supplementary Material The source code for the tool and the tests are on [GitHub:FuzzTree](https://github.com/FuzzTree). The RNA structures encoded as python pickle graphs are available at doi.org/10.5683/SP3/ZR29QE

Funding *Vladimir Reinharz*: NSERC RGPIN-2020-05795, FRQS CBJ1

1 Introduction

The essential regulatory and catalytic roles played by RNAs in cellular processes can largely be attributed to the intriguing and highly versatile nature of their structures [8, 5]. The structure of ncRNAs is inherently modular, with distinct structural domains (loops) divided by stems of rigid canonical bonds, often responsible for their unique functions [20]. This modular architecture has been used for advancements in structure prediction [10] and rational design [11]. Consequently, the characterization of ncRNA structure and identification of



© Théo Boury, Yann Ponty and Vladimir Reinharz; licensed under Creative Commons License CC-BY 4.0

23rd International Workshop on Algorithms in Bioinformatics (WABI 2023).

Editors: Djamel Belazzougui and Aida Ouangraoua; Article No. 12; pp. 12:1–12:22

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

43 structural modules have become critical in the pursuit of understanding their diverse functions
44 and exploiting them for future applications.

45 Many approaches have been developed to detect and classify conserved modules. These
46 classifications differ in the scale adopted to detect and define a motif: RNA3DMotifsAtlas [26]
47 computes similarity and finds motifs at the atomic level. It can capture local similarities
48 omitting bulged nucleotides. A drawback of such a method is the computation time, which
49 restrains comparisons between loops. RNA Bricks [6] and RAG3D [34] abstract loops and
50 hairpins as unitary elements. At an intermediate layer, CaRNAval [27, 30] models RNA as
51 graphs where vertices are nucleotides, and edges are the sequence backbone phosphodiester
52 bonds or non-covalent interactions. These interactions can be classified following the Leontis-
53 Westhof (LW) annotations in 12 different geometric families [21, 31]. Such an approach
54 allows specific graph algorithms to discover much larger and more complex modules than by
55 doing atomic computations while retrieving the known structural modules. However, this
56 approach is not able to identify natural variations since it relies on detecting exact matches.

57 From the algorithmic point of view, the treewidth tw is a natural parameter to find a
58 match of a pattern graph G_P inside a target graph G_T . In 1995, Alon *et al* [1] proposed an
59 XP [9] algorithm in $O(2^{|V_P|} n^{tw(G_P)+1})$ using the color-coding technique. It was shown more
60 generally that only very specific constraints on the input allow having algorithms tractable for
61 bounded treewidths [23]. The problem is not fixed-parameter tractable when parameterized
62 only by the treewidth, and it requires other parameters to become tractable. For instance,
63 some approaches are parameterized both by $tw(G_P)$ and $|G_P|$, and conversely, others are
64 parameterized by $tw(G_T)$ and the maximum degree of G_T [23].

65 However, there can be an exponential number of variants of a specific pattern so different
66 specialized algorithms allowing missing nodes and edges [25, 12], or requiring only labels
67 to be in a neighborhood [18], have been developed. Such simplifications forget about the
68 precise locations of interactions, which is information that we would like to preserve with
69 RNA structures. A recent approach specific to RNA graph fuzziness uses Relational Graph
70 Convolutional Network to embed the graphs in a vector space, allowing fast computation [24].
71 Their embedding is based on the nature of base pairs or their isostericity without taking into
72 account gaps or missing edges. By its nature, such a method gives no explicit control over
73 the sampled neighborhoods, and thresholds need to be calibrated depending on the context.

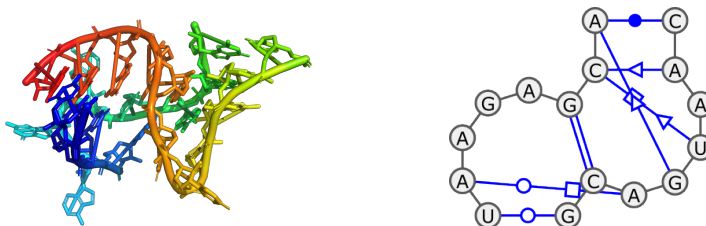
74 In this paper, we introduce FuzzTree, a multidimensional Boltzmann graph sampling
75 procedure able to sample variants of a motif in a known RNA structure. We allow weighting
76 and control of three key geometric features in the variants: (1) the geometric disruption of
77 mismatched edges, (2) missing edges still constrained by their distance in the 3D structure,
78 and (3) breaks in the backbone also constrained by their distance in the 3D structure. We
79 propose a parameterized bound on the complexity of the algorithm based on the treewidth
80 of the searched motif. We evaluate our method on the well-known interior loop Kink-Turn
81 motif [19] characterized by its sharp bend and clustered into 12 different groups in the
82 RNA3DMotifsAtlas [26]. We show that, from the signature representation of the main
83 subgroup, we sample all their known Kink-Turns in 88% of RNAs. We also retrieve two
84 previously un-annotated loops with a characteristic sharp bend.

85 **2 Method**

86 **2.1 RNA as a graph and fundamental problems**

87 We define an RNA structure as a graph G such that its nucleotides are encoded as vertices V ,
88 and nucleotide interactions (canonical/non-canonical base pairs, stacking...) are encoded as

89 directed edges $E \subset V \times V$, with labels $L(e)$. Interactions may represent backbone connectivity
 90 (phosphodiester bonds), or any of the 12 base-pair types defined by the Leontis-Westhof
 91 (LW) nomenclature [31]. Each type specifies an interacting face (Watson-Crick \circ , Hoogsteen
 92 \square , Sugar \triangleright) for both nucleotides, along with an orientation cis (filled) or trans (empty). Note
 93 that the geometry of the RNA structure is encoded in the edge labels, and our representation
 94 does not depend on the sequence. In this work, we are interested in RNA 3D **motifs**, which
 95 we abstract as RNA **pattern graphs** as depicted above. We show in Fig. 1 a Kink-Turn
 96 motif, represented as a graph with labeled edges.



■ **Figure 1 Kink-turn structure.** On the left, the 3D structure of a Kink-Turn motif in PDB 3RW6. On the right, its representation as a pattern graph of its base pair interactions. The backbone connections are represented as black arrowed edges.

97 We rewrite E , the set of edges as $E = B \sqcup \overline{B}$, composed of two distinct sets: B , the set
 98 of edges that are backbone interactions and \overline{B} , the edges involved in LW interactions.

99 Moreover, since vertices in both pattern and target graphs are indexed by their sequence
 100 position, we introduce a precedence relation \prec , inducing a strict total order within the pattern
 101 and target graphs. A valid occurrence of a pattern within a target must be monotonous, *i.e.*
 102 remain consistent with the strict precedence relation \prec .

103 The Monotonous Subgraph Isomorphism (MSI) problem identifies an occurrence of a
 104 pattern $G_P = (V_P, E_P)$ inside a target graph $G_T = (V_T, E_T)$. In the context of RNA, G_P is
 105 a (closed) motif and \prec $-$ Hamiltonian, *i.e.* the total order over V_P induced by the relation
 106 \prec represents a (Hamiltonian) path in G_P , while G_T represents an entire RNA structure.
 107 Formally, the problem of searching for G_P within G_T can be defined as:

108 ► **Problem 1.** *Monotonous Subgraph Isomorphism Problem (MSI)*

109 **Input:** *Pattern graph (\prec $-$ Hamiltonian) $G_P = (V_P, E_P)$; Target graph $G_T = (V_T, E_T)$*

110 **Output:** *Mapping $M : V_P \rightarrow V_T$ such that*

111 ■ $\forall (u, v) \in V_P^2, u \prec v \Rightarrow M(u) \prec M(v)$ (*monotonicity*)

112 ■ $\forall (u, v) \in E_P, (M(u), M(v)) \in E_T \Rightarrow L((u, v)) = L((M(u), M(v)))$ (*label comp.*)

113 ■ $\forall (u, v) \in E_P, (M(u), M(v)) \in E_T$ (*no missing edge*)

114 *or \emptyset if no such mapping exists.*

115 The MSI problem represents a constrained version of Subgraph Isomorphism, a well-studied
 116 NP-complete problem [13, 23] with mildly-depressing prospects regarding parameterized
 117 complexity. Indeed, Subgraph Isomorphism does not admit Fixed-Parameter Tractable (FPT)
 118 or slicewise polynomial (XP) solutions for various graph parameters, including the treewidths
 119 $tw(G_P)$ and $tw(G_T)$ of the pattern and target graphs. Namely, the problem was shown [23]
 120 to be NP-hard even when $\max(tw(G_P), tw(G_T)) \leq 2$ (Para NP-hardness), ruling out the
 121 existence of FPT or XP algorithms under standard hypotheses.

122 The MSI problem retains the classical NP-hardness of Subgraph Isomorphism since it
 123 can be shown to generalize the NP-hard structure-sequence alignment in RNA [28]. However,
 124 MSI can be solved in time $\mathcal{O}(|E_P| \cdot |V_T|^{tw(V_P)+1})$ (XP algorithm) using classic dynamic

125 programming based on a tree decomposition of G_P (see Section 2.5 and Supp. Mat. A.2.2).
 126 Such an algorithm has polynomial complexity for any fixed value of the treewidth $tw(G_P)$, a
 127 parameter that remains bounded in practice (typically 2 or 3) for RNA motifs.

128 2.2 Capturing geometric and chemical similarities

129 We now extend our problem to embrace the natural diversity of RNA motifs in structures.
 130 More precisely, we are interested in sampling graph occurrences that are in the geometric
 131 neighborhood of a core motif. To do so, we allow the motif to be deformed by three different
 132 biologically relevant edit operations detailed below. Each contributes additively and has
 133 its own **neighborhood threshold**, and corresponding difference function, as depicted in
 134 Table 1:

- 135 ■ T^L represents how much we allow the edge label, the type of the canonical or non-canonical
 136 bond, to be modified. It measures the geometric difference between two interactions (see
 137 Sec. 2.4.1).
- 138 ■ T^E corresponds to the maximum number of edges/base pairs within the pattern structure
 139 that can be omitted (see Sec. 2.4.2).
- 140 ■ T^G is the maximum allowed distance when introducing a backbone discontinuity, a new
 141 gap. As insertions alter the distance between bonds, T^G regulates here the maximum
 142 sum over these shifts (see Sec. 2.4.3).

We denote by GEO the **geometric distance** between two nucleotides u_1 and u_2 as

$$\text{GEO}(u_1, u_2) = \min_{a_i \in \text{atoms}(u_i) | i \in \{1,2\}} \|a_1 - a_2\|_2,$$

143 and use it to define two additional criteria to constrain admissible solutions:

- 144 ■ First, nucleotides mapped to the nodes of a missing edge must be closer than $D_{\text{edge}} \text{ \AA}$;
- 145 ■ Second, we enforce a maximal distance D_{gap} between the nucleotide on both sides of
 146 an introduced gap. These values correspond to the phosphodiester atoms' distances
 147 between the nucleotides. Capping these distances beyond a fixed value not only yields
 148 more realistic outputs but also greatly improves the runtime of our algorithm.

149 We use the **isodiscrepancy index** [31] to quantify geometrically the difference between base
 150 pair families and provide values measuring three terms: (1) the difference of intra-base pair
 151 C1'–C1' distances; (2) after aligning one base, the inter-base pair C1'–C1' distance between
 152 the C1' atoms of the second bases of the base pairs; (3) The angle on an axis perpendicular
 153 to the base pair plane required to superpose the second bases. This isostericity measure is
 154 defined for pairs of base pairing families (BPF), each representing one of the 12 canonical
 155 and non-canonical conformations and named as $\text{BPF}_i, \forall i \in \llbracket 1, 12 \rrbracket$. Inter-family variations
 156 are frequent and therefore the average isodiscrepancy of a family to itself is not 0. To correct
 157 for this phenomenon, we define the ISO difference between two families as:

$$158 \quad \text{ISO}(\text{BPF}_i, \text{BPF}_j) = \text{isodiscrepancy}(\text{BPF}_i, \text{BPF}_j) - \text{isodiscrepancy}(\text{BPF}_i, \text{BPF}_i)$$

159 Moreover, we set the value of ISO to 0 involving undefined labels, backbones or phantom
 160 interactions.

161 We define a **backbone path** as a sequence of at least 2 nucleotides connected through
 162 backbone edges.

The set P of paths associated with a target graph $G_T = (V_T, E_T = B_T \sqcup \bar{B}_T)$ is defined
 as:

$$P = \bigcup_{k \in \mathbb{N}, k \geq 2} \{(p_0, \dots, p_k) \mid \forall i \in \llbracket 0, k-1 \rrbracket, (p_i, p_{i+1}) \in B_T\}$$

163 With this definition, gaps are just paths in P with specific restrictions on length and
164 composition.

165 A mapping M lying in a relevant neighborhood of a pattern graph is a solution to a
166 problem that we call the **Fuzzy Monotonous Subgraph Isomorphism problem (FMSI)**,
167 which can be defined as:

168 ► **Problem 2.** *Fuzzy Monotonous Subgraph Isomorphism problem (FMSI)*

169 **Input:** *Pattern graph $G_P = (V_P, E_P = B_P \sqcup \bar{B}_P)$ (\prec -Hamiltonian), target graph $G_T =$
170 $(V_T, E_T = B_T \sqcup \bar{B}_T)$ and neighborhood thresholds $(T^L, T^E, T^G, D_{edge}, D_{gap})$*

171 **Output:** *Mapping $M : V_P \rightarrow V_T$ such that:*

- 172 1. $\forall (u, v) \in V_P^2, u \prec v \Rightarrow M(u) \prec M(v)$ *(monotonicity)*
 - 173 2. $\sum_{(u,v) \in \bar{B}_P} ISO(L(u, v), L(M(u), M(v))) \leq T^L$ *(label compatibility)*
 - 174 3. $\sum_{(u,v) \in \bar{B}_P} 1 - \mathbb{1}_{(M(u), M(v)) \in \bar{B}_T} \leq T^E$ *(few missing edges)*
 - 175 4. $\forall (u, v) \in \bar{B}_P, (M(u), M(v)) \notin \bar{B}_T, GEO(M(u), M(v)) \leq D_{edge}$ *(edge distance limit)*
 - 176 5. $\sum_{(p_0, \dots, p_k) \in P, k \geq 3} GEO(p_0, p_k) \leq T^G$ *(path size limitation)*
 - 177 6. $\forall (u, v) \in B_P, \exists (p_0, p_1, p_2, \dots, p_k) \in P$ such that *(no missing backbone path)*
 - 178 ■ $p_0 = M(u), p_k = M(v)$ *(*)*
 - 179 ■ $GEO(p_0, p_k) \leq D_{gap}$ *(**)*
- 180 or \emptyset if no such mapping exists.

181 Intuitively, a valid mapping M has to respect the six following conditions: The **mono-**
182 **tonicity** condition enforces pattern nodes to map successive nodes in the target. The **label**
183 **compatibility** controls how much the geometric differences cumulative is allowed between
184 pattern and matched edges (see Sec. 2.4.1). The **few missing edges** constraint ensures that
185 pattern edges that are not mapped to an edge in the target are not numerous. (see Sec. 2.4.2)
186 The **edge missing limit** forces each couple of mapped nodes with no edges to have a
187 bounded geometric distance between each other. (see Sec. 2.4.2) The **path size limitation**
188 controls how large the cumulative of gaps geometric lengths can be. (see Sec. 2.4.3) The **no**
189 **missing backbone path** condition (as unfolded in Prob. 2) ensures that the start and
190 end points of a path are mapped nodes (*). It also restrained allowed geometric length of
191 individual path (**). (see Sec. 2.4.3) We note that due to the monotonicity condition, it
192 implies that no target node in p_1, \dots, p_{k-1} can belong to the mapping.

193 Subsequently, we will denote by **neighborhood** $_{G_P}(G_T)$ all the occurrences of the desired
194 pattern graph G_P (in its geometric neighborhood) in our RNA graph target G_T as defined
195 by the previous FMSI mapping.

196 In practice, RNA graphs are fully ordered but do not necessarily contain a Hamiltonian
197 path due to backbone disconnections, leading to a graph composed of multiple strands. We can
198 reconstitute a Hamiltonian path (with no complexity overhead) in the pattern graph by adding
199 some “phantom edges” (with a specific label) when the backbone is missing which correspond
200 to the set of edges $\{(i, i+1) \mid i \in G_P, (i, i+1) \notin E_P \cup L(i, j) \neq \text{”B53”}\}$. Additionally, to
201 ensure that such edge can be mapped in the target G_T in a way that will conserve the monoton-
202 icity of the mapping, we add in G_T the set of edges $\{(i, j) \mid (i, j) \in G_T, i \prec j \cap L(i, j) \neq \text{”B53”}\}$.

203

204 2.3 Locating alternative occurrences through sampling

205 Focusing on neighborhood $_{G_P}(G_T)$ is not an easy task as naive methods would describe both
206 this set and its complementary. In the clique worst case, it consists to explore $\binom{|G_T|}{|G_P|}$ graphs.
207 Even the simple exploration of neighborhood $_{G_P}(G_T)$ can be tedious, in particular, when

12:6 Exploring the natural fuzziness of RNA non-canonical geometries

Threshold T^F	Difference d^F	Fuzzy mapping M of G_P found in G_T
T^L	Isostericity ISO	
T^E	Missing edges number	
T^G	Geometric GEO from 3D structure	

■ **Table 1 Neighborhood thresholds and differences.** Each measure has a threshold over the sum of differences over all edges in the graph pattern.

208 neighborhood thresholds are quite large, which is often the case for label and gap thresholds.
 209 Furthermore, due to the nature of the neighborhoods, numerous instances of a few nucleotides
 210 apart will often be found. It is relevant in terms of neighborhoods, but, from the biological
 211 standpoint, they represent all the same RNA portion and the same underlying geometry and
 212 should not be distinguished: a single representative will be enough. It oriented us toward
 213 sampling, to identify sets of candidate – ideally diverse – subgraphs inside the target graph
 214 G_T that are at a reasonable “ distance” from the interesting motif G_P .

215 This shift in paradigm builds on recent advances in Multidimensional Boltzmann distri-
 216 butions and sampling [2, 15].

217 Generally, a **Boltzmann distribution** is such that the probability of any possible
 218 outcome G depends on its (pseudo-)energy E :

$$219 \quad \mathbb{P}(G) = \frac{e^{-\beta E(G)}}{\mathcal{Z}} \text{ where } \mathcal{Z} = \sum_{G'} e^{-\beta E(G')} \quad (1)$$

where β is a real number, akin to an inverse temperature. A **Multidimensional Boltzmann distribution** (MBD) is a special type of Boltzmann distribution, where the energy is a weighted combination over a collection of features $\{F_i\}$ of interest, such that

$$E(G) = w_1 \times F_1(G) + w_2 \times F_2(G) + \dots$$

220 where $w_1, w_2 \dots$ are real-valued weights. Weights can be used to steer the sampling towards
 221 regions of interest. They can also be learned, through convex optimization, to match
 222 the expectations of $F_1, F_2 \dots$ to user-specified values. Moreover, sampling with a pseudo-

223 temperature $\beta \rightarrow \infty$ gracefully specializes in a uniform random generation of outcomes
 224 achieving optimal (*i.e.* minimal) value for E .

225 In our case, an outcome is a graph $G \subset G_T$, such as G is the image of mapping M and
 226 we have 3 features, one for each neighborhood. Given a specific neighborhood threshold T^F ,
 227 its relative feature F measures how much the weight of edits D^F relative to neighborhoods,
 228 further introduced as a difference in 2.4, deviate from a given center T^{F*} . For instance,
 229 T^{F*} can be chosen as equal to 0 if we want to sample mostly G with no fuzziness or as
 230 equal to $T^F/2$ if we want to sample them with average fuzziness. More details on this choice
 231 and about Boltzmann sampling are available at Supp. Mat. A.1. MBD is well-suited to
 232 the sampling that we want to make: the exponential decrease of the probability with the
 233 features gives low probabilities to the graphs that are far in terms of neighborhoods from
 234 G_P , which allows us to characterize well neighborhood $_{G_P}(G_T)$. In particular, we can define
 235 F such that it takes a value equal to $+\infty$ when the corresponding neighborhood threshold
 236 T^F , for a mapping M , is not respected, forbidding simply M to be sampled. Additionally,
 237 the Multidimensional character of the distribution allows us to take into account the 3
 238 neighborhoods on labels, edges and gaps at the same time.

239 A general framework called **InfraRed** [33], initially introduced in the context of RNA
 240 design [15], can be used to generate efficiently, in a parameterized manner, the MBD. It
 241 automatically processes constraints and elements of the scoring into a graph, decomposes it
 242 into a Tree Decomposition, and generates automatically the bottom-up dynamic programming
 243 sampling procedures. More details on the Tree Decomposition and the dynamic programming
 244 used in **InfraRed** can be found in Supp. Mat. A.2.

2.4 Neighborhood difference description

246 Our goal is to be able to retrieve from a general motif all natural occurrences and their
 247 variability. We can observe in well-known motif families that some bases change, some can
 248 be added or removed. For instance, the graph pattern G_P on Fig. 2 is a Kink-Turn whose
 249 occurrences in the same sub-family can have up to four missing edges. Other sub-families
 250 of Kink-Turn motifs can have differences in bond types, additional interactions, or even
 251 gaps induced by additional nucleotides. We will define difference functions that will be the
 252 features in the MBD and will restrain the samples to a “reasonable” neighborhood of the
 253 pattern G_P that can be explicitly defined.

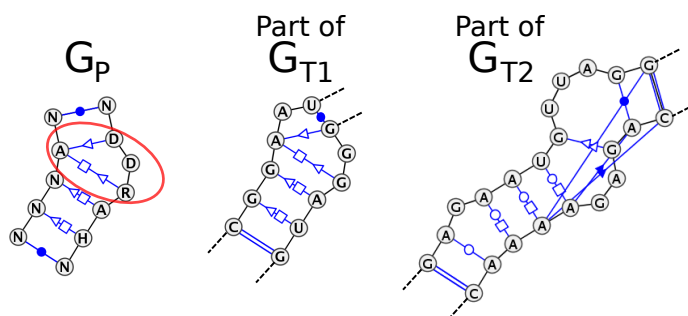
254 For any feature F (here $F \in \{L, E, G\}$, where L are label changes, E missing edges, and
 255 G new gaps) the **Neighborhood cumulative difference** D^F quantifies how distant a
 256 mapping is, relatively to a given neighborhood threshold T^F that cannot be exceeded.

257 Formally, we define a neighborhood cumulative difference D^F relatively to a neighborhood
 258 threshold T^F as:

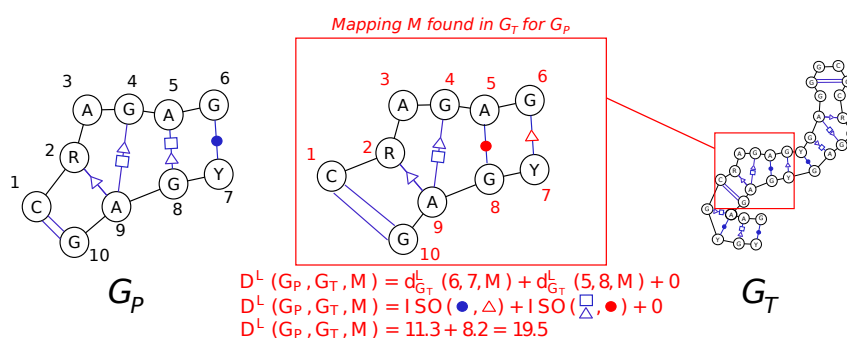
► **Definition 1** (Neighborhood cumulative difference / neighborhood difference). *Given a
 pattern graph $G_P = (V_P, E_P = B_P \sqcup \bar{B}_P)$, a target graph $G_T = (V_T, E_T = B_T \sqcup \bar{B}_T)$
 and a mapping M , a neighborhood cumulative difference is a function D^F relatively to a
 neighborhood threshold T^F that act as a wrapper around $d_{G_T}^F$:*

$$D^F(G_P, G_T, M) = \sum_{(u,v) \in E_P} d_{G_T}^F(u, v, M)$$

259 where $d_{G_T}^F(u, v, M)$ is the **neighborhood difference** relative to G_T , a function that measures,
 260 relatively to F , how “different” are the edges in the pattern $((u, v) \in G_P)$ from the edges in
 261 the mapping $((M(u), M(v)) \in G_T)$.



■ **Figure 2 Kink-turn signature and targets.** On the left, signature graph of the Kink-Turn IL_29549.9 family and our search pattern. In the middle and on the left, mappings that were missed during the search for the pattern. G_{T1} due to the same nucleotide merging the end of a cSS and a cWW. G_{T2} due to its too large difference.



■ **Figure 3 Label difference.** Computation of the label difference on a mapping between a motif G_P and an RNA target graph G_T . Label difference is computed using the isostericity ISO to account for the geometric difference between bounds as described in Stombaugh et al [31].

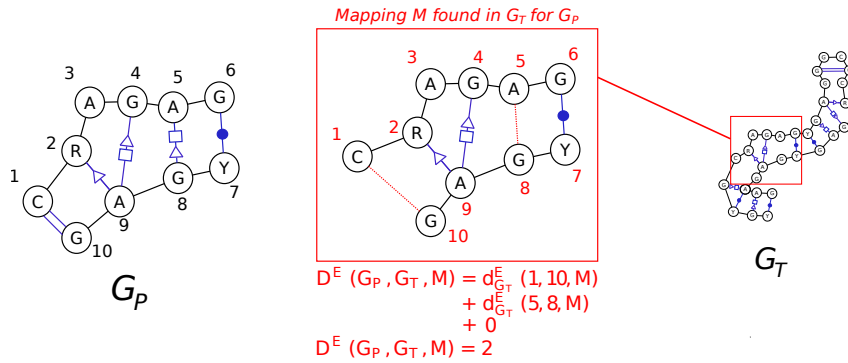
262 How the difference is measured depends on the feature as described below.

263 Neighborhood cumulative differences serve in the Boltzmann distribution to quantify
 264 each type of edit. Due to the additivity of these deformations, the neighborhood cumulative
 265 differences are computed over all edges in the pattern and their equivalent in the mapping.
 266 While our neighborhood cumulative differences are defined relative to the edges of G_P here,
 267 they can be easily defined on nodes should novel sequence-dependant features be included.
 268 We will now discuss in detail the 3 sources of operations and their neighborhood cumulative
 269 difference. A summary is shown in Table 1.

270 2.4.1 The label difference

271 The label difference, as represented in Fig. 3, accounts for the difference between base
 272 pairs families and we use for that the isodiscrepancy [31] as introduced in part 2.2. We now
 273 compute the label difference D^L relative to the neighborhood threshold T^L as a neighborhood
 274 cumulative difference entirely defined by the sum over each pattern edge of its mapping
 275 neighborhood difference $d_{G_T}^L$ equals to:

$$d_{G_T}^L(u, v, M) = \text{ISO}(L(u, v), L(M(u), M(v)))$$



■ **Figure 4 Edge difference.** Computation of the edge difference on a mapping between a motif G_P and an RNA target graph G_T . We assume here that $D_{edge} \gg \max(\text{GEO}(1, 10), \text{GEO}(5, 8))$

2.4.2 The edge difference

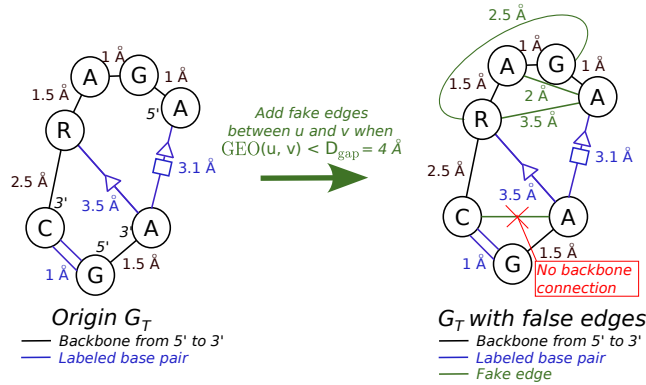
While the previous section deals with how to incorporate edges changing their type, *i.e.* their interaction geometry, we must also consider that some of these base interactions might simply be missing due to the noisiness of the experiments, the accuracy of the annotation, or the flexibility of the module. A natural way to account for missing edges is to count them and enforce an upper bound on the amount. Doing so would omit important geometric information that we have available in the 3D structure. An interaction is missing, but we still want to constrain the physical distance between the mapped nodes of the missing edge. Indeed, with no limitation on that distance, the partner node of a missing edge could be virtually anywhere in the target structure. This is undesirable since we are interested in patterns matching the local conformations. It is also highly inefficient in terms of computation.

Therefore, we will accept mappings of the extremities of an edge in the pattern to nodes u, v that are at most at a set threshold distance D_{edge} computed from the 3D structure (*i.e.* $\text{GEO}(u, v) < D_{edge}$). Setting a weight of ∞ to mappings outside the threshold allows the sampling to simply reject such instances. We additionally use the edge difference to reject cases where backbones are mapped to couples of nodes that are not backbones by putting a weight ∞ in that case. The total edge difference D^E relative to neighborhood threshold T^E , is a neighborhood cumulative difference entirely defined by the sum over $d_{G_T}^E$ with values defined as followed and shown in Fig. 4:

$$d_{G_T}^E(u, v, M) = \begin{cases} 0 & \text{if } (u, v) \in B_P \cap (M(u), M(v)) \in B_T \\ & \text{or } (u, v) \in \bar{B}_P \cap (M(u), M(v)) \in \bar{B}_T \\ 1 & \text{if } (u, v) \in \bar{B}_P \cap (M(u), M(v)) \notin \bar{B}_T \\ & \text{and } \text{GEO}(M(u), M(v)) \leq D_{edge} \\ \infty & \text{otherwise} \end{cases}$$

2.4.3 The gap difference

A frequent type of natural variability in a motif family is the occurrence of bulging out nucleotides in what would be a continuous sequence in the pattern. These insertions can be of different sizes, but we require that they do not modify (too much) the local structure. To take arbitrary insertions into account, we introduce **fake edges** between any two nucleotides present on the same backbone that are at a distance below D_{gap} . An illustration of this



■ **Figure 5 Fake edges.** Addition of fake edges to account for gaps. Fake edges are added only when distance is below D_{gap} and when both nucleotides are fully connected by backbone edges. For instance here, we add no fake edge between C and A at the bottom of G_T as these two nucleotides are not connected by a full path of backbones.

293 process is shown in Fig. 5. For convenience, these edges are added in B_T to keep valid the
 294 cases of the edge difference where backbones are wrongly mapped.

295 An additional difference compared to the missing interaction edges of the previous section
 296 is how we sum the total neighborhood difference D^G . We accumulate the total physical
 297 distance (*i.e.* GEO) between the nodes connected through the fake edges. This allows an
 298 arbitrarily large structure to bulge out without the need to verify or specify admissible lengths,
 299 as long as the nucleotides around this inserted gap are close geometrically as illustrated in
 300 Fig. 6.

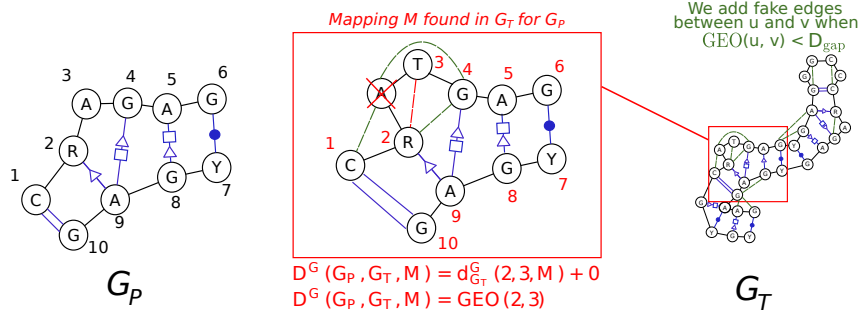
Formally, the gap difference D^G relative to neighborhood threshold T^G is a neighborhood
 cumulative difference over all edges in the matching entirely defined by the sum of the
 neighborhood differences $d_{G_T}^G$:

$$d_{G_T}^G(u, v, M) = \begin{cases} \text{GEO}(M(u), M(v)) & \text{if } (M(u), M(v)) \text{ is} \\ & \text{a "Fake Edge" in } E_T \\ 0 & \text{otherwise} \end{cases}$$

301 A limitation of this approach is that we cannot detect the deletion of nodes from the
 302 pattern. A workaround is to remove all the nodes in the pattern graph that do not directly
 303 participate in a base pair interaction, and reconnect the disconnected backbones. Using the
 304 new pattern with a large gap threshold T^G would allow us to retrieve the original motif
 305 neighborhood efficiently, but introduce more spurious matches.

306 2.5 Algorithm and complexity

307 Our method is based on Infrared [15, 33], a declarative framework that automatically
 308 generates a dynamic programming procedure for MBD sampling, based on a nice tree
 309 decomposition (TD). The dynamic programming procedure used in Infrared is described
 310 in Supp. Mat. A.2. It precomputes the partition function of the MBD through a bottom-
 311 up recursion and uses local contributions to perform an exact sampling within the MBD
 312 distribution. Within this framework, a combinatorial problem is abstracted as a set of
 313 variables $\{X_i\}_i$, each assigned an integer value within a bounded domain. Assignments
 314 must respect various constraints expressed as functions $\{C_i\}_i$, each defined over a subset of

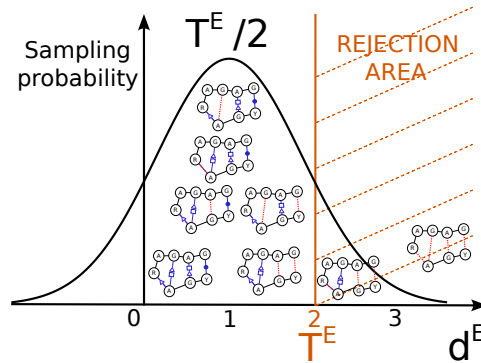


■ **Figure 6 Gap difference.** Computation of gap difference on a mapping between a motif G_P and an RNA target graph G_T . We recall that nucleotide labels are not taken into account.

315 variables. Similarly, feature functions $\{F_j\}_j$ associate real-valued contributions to subsets of
 316 variables, and are summed to represent the pseudo-energy of an assignment.

317 In this setting, we abstract each node i of the graph pattern G_P as a variable X_i , taking
 318 value in $\llbracket 1, n \rrbracket$. The value of X_i represents the mapping of node i in the graph $G_T = (V_T, E_T)$
 319 with $|V_P| = k$ and $|V_T| = n$. Within RNA motifs, the number of partners of a position is
 320 bounded, so we have $|E_P| \in \mathcal{O}(k)$. Remark that all deviations from the pattern defined in
 321 Sections 2.4.1 through 2.4.3, can be expressed *locally* as sums on the edges of the pattern
 322 graph. It follows that the dependencies dep implied by our cumulative differences are only
 323 binary, and restricted to pairs sharing an edge in G_P : $dep = \{(X_i, X_j) \mid (i, j) \in E_P\}$. The
 324 graph of constraints is thus reducible to the input pattern graph G_P , as shown in Fig. 8.

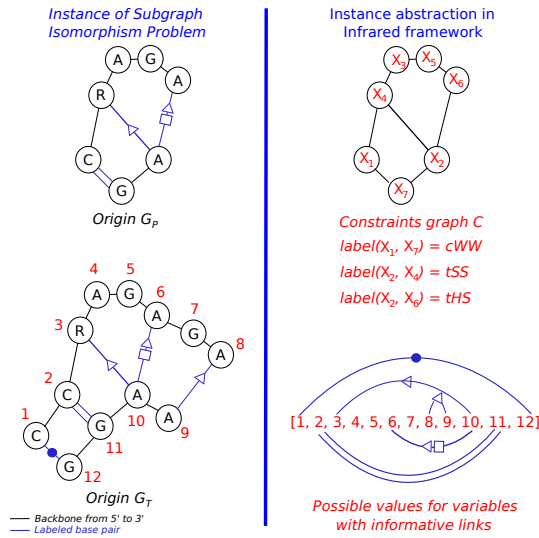
325 Due to the neighborhood threshold T^F being a global property over the mapping, the
 326 sampling is followed by a rejection step for samples that exceed a neighborhood threshold.
 327 An example of such rejection is depicted in Fig. 7. Asymptotically, such rejection will at
 328 worst induce a constant overhead with T^F chosen independently from $|G_P|$ and $|G_T|$.



■ **Figure 7 Rejection step.** In the above example, rejection is depicted only for the edge neighborhood for the sake of simplicity. Found motifs above T^E thresholds are rejected afterward. Found motifs with an edge difference close to $\frac{T^E}{2} = 1$ here have more chance to be sampled.

329 ► **Proposition 1.** A generation of t Boltzmann-distributed (1) putative solutions to FMSI
 330 can be performed in time $\mathcal{O}(nkt + kn^{(\phi+1)})$ where ϕ is the treewidth of the pattern G_P .

331 This complexity directly follows from the complexity of the algorithm [15] underlying
 332 **Infrared** for a graph $G_P = (V_P, E_P)$ (with $|V_P| = k$). Restricted to binary constraints/features
 333 associated with (a subset of) E , the computation of the partition function can be



■ **Figure 8 Framework abstraction.** Interfacing Infrared by considering G_P as the Infrared graph of constraints C and all nodes of G_T as values that can be taken by the variables in C .

334 performed in time $\mathcal{O}((|E_P| + |V_P|) \times \Delta^{\phi+1})$, where Δ is the size of the assignment domain for
 335 individual variables, and ϕ is the treewidth of G_P . A stochastic backtrack follows, leading to
 336 the generation of t Boltzmann-distributed assignments in time $\mathcal{O}(|V_P| \Delta t)$. The complexity
 337 stated above is obtained by observing that $|E_P| \in \Theta(k)$, and that $\Delta \in \Theta(n)$.

338 We conclude by noting that preprocessing, including computations of geometrical distances
 339 and augmentation of G_T graph, can be performed once, in $O(n^2)$ time and space, leading to
 340 a negligible overhead in comparison to the computation of the partition function. Meanwhile,
 341 an optimal tree decomposition can be theoretically obtained in time only super polynomial
 342 in ϕ [3].

343 A summary of the complexity and capacity of our FuzzTree method is depicted in Table 2.
 344 Regarding the parameterized complexity [9], the FuzzTree method is XP in the treewidth of
 345 the pattern graph, both in time and in space. It represents progress compared to VF2 [7],
 346 which is indeed implemented and efficient in practice due to the profusion of lookahead rules
 347 but has a worst-case time complexity similar to $O(n^n)$. In practice, VF2 becomes costly
 348 with dense graphs, even in its most modern versions [4, 17]. Furthermore, we compete with
 349 the bound from the Color-Coding [1] technique by improving it in time and space. $2^{O(k)}$ is
 350 replaced by $k \leq n$ in our bounds, which allows us to get rid of k as a parameter to restrict it
 351 simply to the treewidth in our RNA case.

352 In addition, our method handles at the same time multiple labels on edges, directed
 353 graphs and can integrate node labels. The latter has not been implemented but can be
 354 added, as with labels on edges, without complexity overhead.

355 3 Results

356 3.1 Computations

The larger target graphs (of more than 500 nucleotides) were split into overlapping voxels to increase computational efficiency. We extracted $|G_T|$ graphs centered in each nucleotide c at

Method Name	Color-Coding [1]	VF2 [7]	VeRNAI [24]	FuzzTree
Year	1995	2004 (updated up to 2018)	2021	2022
Method	Tree coloring	DFS with search space reduction	Relational Graph Convolution Network	Sampling technique
Time complexity	$2^{O(k)}n^{\phi+1}\log(n)$	$O(\deg(G_T)^n)$	Exponential	$O(knt + kn^{\phi+1})$
Space complexity	$2^{O(k)}n^{\phi+1}$	$O(n)$	Exponential	$O(n^{\phi+1})$
Supported graph	Directed and undirected	Undirected	Directed and undirected	Directed and undirected
Supported labels	One label by edge	One label by node	Any number of labels on edges and nodes	Any number of labels on edges and nodes
Type of found neighborhoods	None	None	Isostericity related	Exact bound on isostericity, missing edge and missing gap.
Implementation?	No	Yes	Yes	Yes

■ **Table 2 Complexities for RNA motif search.** Comparison of state-of-the-art methods for RNA motif search. With $\phi = tw(G_P)$, $n = |V_T|$, $k = |V_P|$ and t the number of samples.

a given radius R from c . For an extracted graph G , centered on c , we have :

$$\forall j \in G, R(G) = \text{GEO}(j, c) \leq R$$

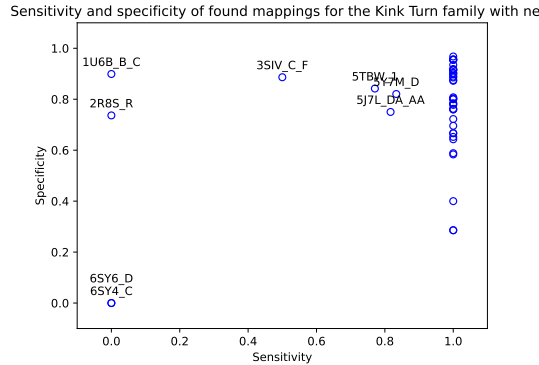
357 Choices of technical parameters, such as the value for R , hardware and computation times are
 358 discussed in Supp. Mat. A.3. For the sake of efficiency, we refrained from adding "phantom
 359 edges" described in Section 2.2. Doing so enables possible violations of the monotonicity,
 360 leading to the detection of motif occurrences in the context of a more remote homology, but
 361 necessitated a further round of rejection (whose impact on performances remained negligible).

362 3.2 Data: the Kink-Turn motifs family

363 All interactions in the RNA structures are provided by FR3D [29]. We also use interactions
 364 annotated as "near". The Kink-Turn is an important RNA structural motif common in duplex
 365 RNA that creates a sharp axial bend, enabling crucial tertiary interactions and binding [19].
 366 The Kink-Turn has been shown to appear in multitudes of contexts through computational
 367 and experimental methods [16, 22]. As of January 2023, there were 72 instances of the
 368 Kink-Turn RNA annotated in the RNA3DMotifAtlas [26]. One was omitted because it was
 369 not annotated on the main structure but one of its symmetric alternatives. The others span
 370 46 different RNAs and are divided into 12 different families with different lengths, between 9
 371 and 23 nucleotides and base pair signature. Members of the same family also differ in terms
 372 of number of nucleotides and pairing.

373 The Kink-Turn family IL_29549.9 in RNA3DMotifsAtlas has the most occurrences (32)
 374 and its signature graph shown in Fig. 2 is used as the pattern graph G_P for the subsequent
 375 sampling.

376 Empirically, RNA 3D motifs are small motifs that, despite not being tree-like, have
 377 relatively small treewidth. It is especially the case for the Kink-Turn family, where 50
 378 Kink-Turns pattern graphs have treewidth equal to 2 and 21 have treewidth equal to 3, which
 379 makes our parameterization in treewidth practically quite relevant.



■ **Figure 9 Sensitivity and Specificity of regions corresponding to sampled graphs in the 46 RNA structures containing Kink-Turns.** Each dot represents an RNA chain, where one or multiple Kink-Turns can be found. To keep track of them, nodes whose sensitivity is not equal to one, are named of the graph “RNAname”_“chain”.

380 3.2.1 Results

381 We use the parameters shown in Table 3 with G_P in Fig. 2 to sample at least 1000 graphs
 382 in each of the 46 RNA structures. We also introduce a bias in the Boltzmann distribution
 383 to favor values of neighborhood thresholds equal to $\frac{T^E}{2}$ (instead of 0) to favor slightly
 384 fuzzy mappings more often than exact mappings or extremely fuzzy ones. This choice is
 385 motivated by the focus on the neighborhood more than on the exact mappings for which lots
 386 of techniques already exist.

Parameter	T^L	T^E	T^G	D_{edge}	D_{gap}	R	nb_samples
Used value	20.0	4	20.0	5.0	10.0	$R(G_P) + \frac{D_{\text{gap}}}{4}$	1000
Relevant range	[0, 50]	[[0, 6]]	[0, 50]	[5, 10]	[5, 20]	$R(G_P) + [\frac{D_{\text{gap}}}{4}, D_{\text{gap}}]$	

■ **Table 3 Parameters.** Used parameters and relevant range for FuzzTree computation on the Kink-Turn group.

387 Our sampling returns sub-graphs of the target graphs G_T . Using a python implementation
 388 of VF2 [14, 7], we annotate in the 46 RNAs graphs all nucleotides in any of the mappings.
 389 Each of the connected components in the 46 RNAs becomes a hit. The True Positives (TP)
 390 are these covering a known Kink-Turn found by our method. The True Negative (TN) are
 391 those that do not cover a Kink-Turn, rightly not found by our method. P designs the set of
 392 all Kink-Turn motifs and N the set of all other motifs. We show the sensitivity (TP/P) and
 393 specificity (TN/N) per RNA structure in Fig. 9.

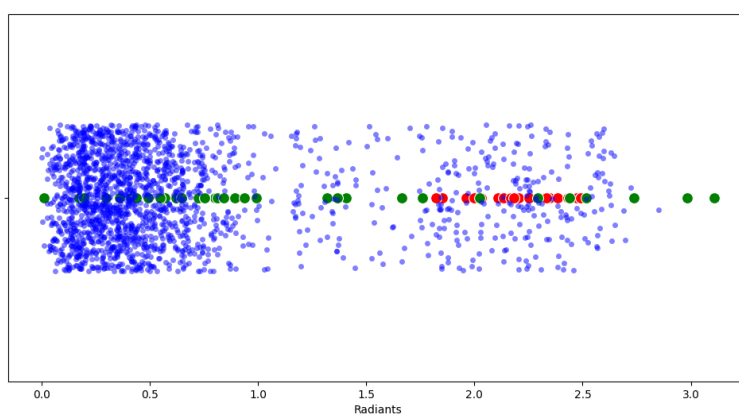
394 In 38 out of the 46 RNAs a sensitivity of 1 is achieved, all Kink-Turns are covered in
 395 graphs sampled by our method. The missing Kink-Turns fall into two categories. First,
 396 too many missing edges: with only 6 Leontis-Westhof interactions in G_T , allowing more
 397 missing edges would match any interaction in the targets. Second, backbone connections
 398 replaced by Leontis-Westhof interactions, as seen in the middle of Fig. 2, is not an allowable
 399 transformation in our model.

400 We also obtain in 33 RNAs a specificity over 75%. It indicates that even with relatively
 401 lax parameters, not that many other instances in comparison to the amount of known

402 Kink-Turns are close to G_T .

403 3.2.2 Other identified regions

404 An additional 198 locations in the 46 RNAs were identified. The Kink-Turn is essentially an
 405 internal loop motif. We investigate if other internal loops sharing the same main 3D feature, a
 406 sharp bend in an interior loop, are found. Using the python library forgi [32] we decomposed
 407 these regions in their secondary structure elements. The majority, 125, mapped to regions
 408 forming multiloops. A total of 33 were covering continuous double-stranded regions. The
 409 angles of surrounding stems for each interior loop in the 46 RNAs (in blue) the identified
 410 Kink-Turns in these RNAs (red) and the other 33 elements (in green) are shown in Fig. 10.



■ **Figure 10 Angles in radians.** In blue for stems around every interior loop in the 46 RNAs. In red for the Kink-Turns identified in these RNAs. In green for the additional 33 continuous double-stranded regions.

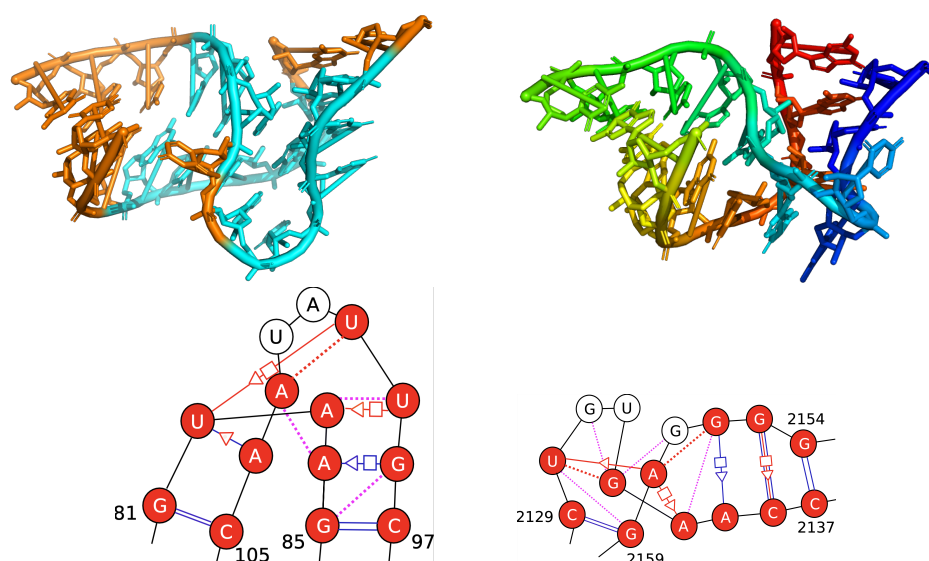
411 There are 10 additional regions with angles above 1.4rad, and two of these had a sharp
 412 turn in their structure in un-annotated region as seen in Fig. 11. We show below their graph
 413 of interactions, with the cross-strand stackings in orange.

414 The first is in 5J7L chain DA and positions 78–86, 96–108. It overlaps an un-annotated
 415 motif (IL_85931.1) that covers positions 81–85, 97–101, and 103–105. The second is located
 416 in 7RQB, chain 1A, positions 2129–2138, 2153–2160, and is not covered or surrounded by
 417 any annotated motif.

418 4 Conclusion

419 In this paper, we introduce FuzzTree, a multidimensional Boltzmann method for sampling a
 420 graph pattern neighborhood in a target graph. FuzzTree defines three types of neighborhoods
 421 based on RNA geometric diversity, LW interaction modifications, missing edges, and breaks
 422 in the backbone. Each can be explicitly controlled. We show that our sampling method
 423 complexity is parameterized by the treewidth of the pattern graph.

424 Two main limitations are inherent to our approach. Due to the intrinsic nature of
 425 sampling, we cannot be assured that all neighboring graphs will be reported. In itself, for
 426 large patterns, this is a feature since sampling allows uniform exploration of the exponentially
 427 growing neighborhood. By enabling per-feature biases, FuzzTree can also be calibrated to



■ **Figure 11 Other matches.** 5J7L on the left and 7RQB on the right. The 3D structure on the left has IL_85931.1 highlighted in cyan, on the right each nucleotide is colored independently. In the graphs, red nodes are matched with the pattern. Blue edges are in the RNA structure and red ones are in the pattern, indicating modifications and removal. Red dashed lines are introduced “Fake edges”. Magenta dashed lines indicate stackings.

428 favor the sampling of graphs at a desired location in the neighborhood to favor specific types
 429 of variants (e.g., isostericity of modified edges). Letting the sampling run for longer will also
 430 mitigate the problem. More importantly, some patterns cannot be identified, particularly if
 431 an LW interaction is replaced by a backbone connection. While such cases are rare, they do
 432 exist, and additional improvement will be needed to capture them.

433 We evaluate our method on the Kink-Turn group, a well-known interior loop motif that
 434 induces a sharp bend in the structure and is annotated in 46 different RNA structures.
 435 The Kink-Turns are grouped in the RNA3DMotifAtlas into 12 different subgroups with
 436 varying lengths and interactions. Using only the signature graph of one subgroup, FuzzTree
 437 samples conformations of over 2/3 of all Kink-Turns and identifies all of them in 88% of RNA
 438 structures. A closer examination of the other sampled patterns reveals two previously un-
 439 annotated sub-structures, each with a characteristic G-A trans-Hoogsteen-sugar interaction
 440 and a sharp local bend.

441 Future work to complement this should broaden the evaluation framework by testing
 442 FuzzTree on diverse RNA modules. There is also a need for new techniques to overcome
 443 pattern identification limitations and explore adaptive sampling strategies to dynamically
 444 steer the sampled neighborhood.

445 While FuzzTree was developed and adapted for RNA structure modules, it highlights the
 446 flexibility of multidimensional Boltzmann sampling and could be applied to other biological
 447 networks such as protein-protein interaction networks or metabolic pathways. Addressing
 448 these questions and areas for future work could lead to more comprehensive insights into
 449 complex RNA structures and other biological networks.

References

- 450 1 Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *J. ACM*, 42(4):844–856, jul 1995.
451 [doi:10.1145/210332.210337](https://doi.org/10.1145/210332.210337).
- 452 2 Olivier Bodini and Yann Ponty. Multi-dimensional Boltzmann Sampling of Languages. *Dis-*
453 *crete Mathematics & Theoretical Computer Science*, DMTCS Proceedings vol. AM, 21st
454 International Meeting on Probabilistic, Combinatorial, and Asymptotic Methods in the Ana-
455 lysis of Algorithms (AofA'10), January 2010. URL: <https://dmtcs.episciences.org/2793>,
456 [doi:10.46298/dmtcs.2793](https://doi.org/10.46298/dmtcs.2793).
- 457 3 Hans L. Bodlaender and Arie M. C. A. Koster. Combinatorial optimization on graphs of
458 bounded treewidth. *The Computer Journal*, 51(3):255–269, 2008. [doi:10.1093/comjnl/
459 bxm037](https://doi.org/10.1093/comjnl/bxm037).
- 460 4 Vincenzo Carletti, Pasquale Foggia, Alessia Saggese, and Mario Vento. Introducing vf3: A
461 new algorithm for subgraph isomorphism. In Pasquale Foggia, Cheng-Lin Liu, and Mario
462 Vento, editors, *Graph-Based Representations in Pattern Recognition*, pages 128–139, Cham,
463 2017. Springer International Publishing.
- 464 5 Thomas R Cech and Joan A Steitz. The noncoding RNA revolution—trashing old rules to
465 forge new ones. *Cell*, 157(1):77–94, 2014.
- 466 6 G Chojnowski, T Waleń, and JM Bujnicki. RNA Bricks—a database of RNA 3D motifs and their
467 interactions. *Nucleic Acids Research*, 42, 2013. URL: <https://doi.org/10.1093/nar/gkp011>,
468 [doi:10.1093/nar/gkt1084](https://doi.org/10.1093/nar/gkt1084).
- 469 7 Luigi Cordella, Pasquale Foggia, Carlo Sansone, and Mario Vento. A (sub)graph isomorphism
470 algorithm for matching large graphs. *Pattern Analysis and Machine Intelligence, IEEE*
471 *Transactions on*, 26:1367 – 1372, 11 2004. [doi:10.1109/TPAMI.2004.75](https://doi.org/10.1109/TPAMI.2004.75).
- 472 8 José Almeida Cruz and Eric Westhof. The dynamic landscapes of RNA architecture. *Cell*,
473 136(4):604–609, 2009.
- 474 9 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Daniel Marx, Marcin
475 Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2016.
- 476 10 Rhiju Das, Rachael C Kretsch, Adam J Simpkin, Thomas Mulvaney, Phillip Pham, Ramya
477 Rangan, Fan Bu, Ronan Keegan, Maya Topf, Daniel Rigden, et al. Assessment of three-
478 dimensional RNA structure prediction in CASP15. *bioRxiv*, pages 2023–04, 2023.
- 479 11 Sven Findeiß, Christoph Flamm, and Yann Ponty. Rational Design of RiboNucleic Acids
480 (Dagstuhl Seminar 22381). *Dagstuhl Reports*, 12(9):121–149, 2023. URL: [https://drops.
481 dagstuhl.de/opus/volltexte/2023/17811](https://drops.dagstuhl.de/opus/volltexte/2023/17811), [doi:10.4230/DagRep.12.9.121](https://doi.org/10.4230/DagRep.12.9.121).
- 482 12 Nagoor Gani. 63. isomorphism on fuzzy graphs. *International Journal of Computational and*
483 *Mathematical Sciences*, Vol. 2:200–206, 01 2008. [doi:10.13140/2.1.1873.9847](https://doi.org/10.13140/2.1.1873.9847).
- 484 13 M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of*
485 *NP-Completeness*. W. H. Freeman, 1979.
- 486 14 Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and
487 function using NetworkX. Technical report, Los Alamos National Lab.(LANL), Los Alamos,
488 NM (United States), 2008.
- 489 15 S. Hammer, W. Wang, S. Will, and Y. Ponty. Fixed-parameter tractable sampling for rna
490 design with multiple target structures. *BMC Bioinformatics*, 2019.
- 491 16 Lin Huang and David MJ Lilley. The kink turn, a key architectural element in RNA structure.
492 *Journal of molecular biology*, 428(5):790–801, 2016.
- 493 17 Alpár Jüttner and Péter Madarasi. Vf2++—an improved subgraph isomorphism algorithm. *Dis-*
494 *crete Applied Mathematics*, 242:69–81, 2018. Computational Advances in Combinatorial Optim-
495 ization. URL: <https://www.sciencedirect.com/science/article/pii/S0166218X18300829>,
496 [doi:https://doi.org/10.1016/j.dam.2018.02.018](https://doi.org/10.1016/j.dam.2018.02.018).
- 497 18 Arijit Khan, Nan Li, Xifeng Yan, Ziyu Guan, Supriyo Chakraborty, and Shu Tao. Neighborhood
498 based fast graph search in large networks. In *Proceedings of the 2011 ACM SIGMOD*
499 *International Conference on Management of Data*, SIGMOD '11, page 901–912, New York,
500 NY, USA, 2011. Association for Computing Machinery. [doi:10.1145/1989323.1989418](https://doi.org/10.1145/1989323.1989418).
- 501

- 502 19 Daniel J Klein, T Martin Schmeing, Peter B Moore, and Thomas A Steitz. The kink-turn: a
503 new RNA secondary structure motif. *The EMBO journal*, 20(15):4214–4221, 2001.
- 504 20 Neocles B Leontis, Aurelie Lescoute, and Eric Westhof. The building blocks and motifs of
505 RNA architecture. *Current Opinion in Structural Biology*, 16(3):279–287, 2006. Nucleic
506 acids/Sequences and topology. URL: [https://www.sciencedirect.com/science/article/
507 pii/S0959440X06000807](https://www.sciencedirect.com/science/article/pii/S0959440X06000807), doi:<https://doi.org/10.1016/j.sbi.2006.05.009>.
- 508 21 Neocles B Leontis and Eric Westhof. Geometric nomenclature and classification of RNA base
509 pairs. *Rna*, 7(4):499–512, 2001.
- 510 22 Bin Li, Shurong Liu, Wujian Zheng, Anrui Liu, Peng Yu, Di Wu, Jie Zhou, Ping Zhang, Chang
511 Liu, Qiao Lin, et al. RIP-PEN-seq identifies a class of kink-turn RNAs as splicing regulators.
512 *Nature Biotechnology*, pages 1–13, 2023.
- 513 23 Dániel Marx and Michal Pilipczuk. Everything you always wanted to know about the
514 parameterized complexity of Subgraph Isomorphism (but were afraid to ask). In Ernst W.
515 Mayr and Natacha Portier, editors, *31st International Symposium on Theoretical Aspects of
516 Computer Science (STACS 2014)*, volume 25 of *Leibniz International Proceedings in Informatics
517 (LIPIcs)*, pages 542–553, Dagstuhl, Germany, 2014. Schloss Dagstuhl–Leibniz-Zentrum fuer
518 Informatik. URL: <http://drops.dagstuhl.de/opus/volltexte/2014/4486>, doi:10.4230/
519 LIPIcs.STACS.2014.542.
- 520 24 Carlos Oliver, Vincent Mallet, Pericles Philippopoulos, William L Hamilton, and Jérôme
521 Waldispühl. Vernal: a tool for mining fuzzy network motifs in RNA. *Bioinformatics*, 38(4):970–
522 976, 11 2021. arXiv:[https://academic.oup.com/bioinformatics/article-pdf/38/4/970/
523 42319124/btab768.pdf](https://academic.oup.com/bioinformatics/article-pdf/38/4/970/42319124/btab768.pdf), doi:10.1093/bioinformatics/btab768.
- 524 25 Aymeric Perchant and Isabelle Bloch. Fuzzy morphisms between graphs. *Fuzzy Sets and
525 Systems*, 128(2):149–168, 2002. URL: [https://www.sciencedirect.com/science/article/
526 pii/S0165011401001312](https://www.sciencedirect.com/science/article/pii/S0165011401001312), doi:[https://doi.org/10.1016/S0165-0114\(01\)00131-2](https://doi.org/10.1016/S0165-0114(01)00131-2).
- 527 26 Anton I. Petrov, Craig L. Zirbel, and Neocles B. Leontis. Automated classification of rna 3d
528 motifs and the rna 3d motif atlas. *RNA*, 2013.
- 529 27 Vladimir Reinharz, Antoine Soulé, Eric Westhof, Jérôme Waldispühl, and Alain Denise. Mining
530 for recurrent long-range interactions in RNA structures reveals embedded hierarchies in network
531 families. *Nucleic Acids Research*, 46(8):3841–3851, 03 2018. arXiv:[https://academic.oup.
532 com/nar/article-pdf/46/8/3841/24783244/gky197.pdf](https://academic.oup.com/nar/article-pdf/46/8/3841/24783244/gky197.pdf), doi:10.1093/nar/gky197.
- 533 28 Philippe Rinaudo, Yann Ponty, Dominique Barth, and Alain Denise. Tree decomposition and
534 parameterized algorithms for rna structure-sequence alignment including tertiary interactions
535 and pseudoknots. In Ben Raphael and Jijun Tang, editors, *Algorithms in Bioinformatics*,
536 pages 149–164, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.
- 537 29 Michael Sarver, Craig L Zirbel, Jesse Stombaugh, Ali Mokdad, and Neocles B Leontis. FR3D:
538 finding local and composite recurrent structural motifs in RNA 3D structures. *Journal of
539 mathematical biology*, 56:215–252, 2008.
- 540 30 Antoine Soulé, Vladimir Reinharz, Roman Sarrazin-Gendron, Alain Denise, and Jérôme
541 Waldispühl. Finding recurrent RNA structural networks with fast maximal common subgraphs
542 of edge-colored graphs. *PLoS computational biology*, 17(5):e1008990, 2021.
- 543 31 Jesse Stombaugh, Craig L. Zirbel, Eric Westhof, and Neocles B. Leontis. Frequency and isoster-
544 icity of RNA base pairs. *Nucleic Acids Research*, 37(7):2294–2312, 02 2009. arXiv:[https://
545 academic.oup.com/nar/article-pdf/37/7/2294/18783207/gkp011.pdf](https://academic.oup.com/nar/article-pdf/37/7/2294/18783207/gkp011.pdf), doi:10.1093/nar/
546 gkp011.
- 547 32 Bernhard C Thiel, Irene K Beckmann, Peter Kerpedjiev, and Ivo L Hofacker. 3D based on
548 2D: Calculating helix angles and stacking patterns using forgi 2.0, an RNA Python library
549 centered on secondary structure elements. *F1000Research*, 8, 2019.
- 550 33 Hua-Ting Yao, Yann Ponty, and Sebastian Will. Developing complex rna design applications
551 in the infrared framework. *RNA Folding - Methods and Protocols*, 2022.

- 552 34 M. Zahran, C. Sevim Bayrak, S. Elmetwaly, and T. Schlick. RAG-3D: a search tool for RNA
 553 3D substructures. Nucleic acids research. *Nucleic Acids Research*, 43(19):9474–9488, 2015.
 554 doi:<https://doi.org/10.1093/nar/gkv823>.

555 **A** Supplementary material

556 **A.1** About the sampling process

557 Sampling from a Multidimensional distribution in our case can be written formally as below
 558 :

► **Definition 2** (Boltzmann distribution/Partition function). *In the **Multidimensional Boltzmann Distribution**, the probability to sample graph G , subgraph of G_T with features F_1, \dots, F_m (that embody neighborhoods differences of G_P for mapped graph G in G_T) of respective weights w_1, \dots, w_m (that we can write more simply $w = (w_1, \dots, w_m)$) is proportional to its **energy**:*

$$\mathbb{P}_{G_P, G_T}(G \mid w) = \frac{\prod_{i=1}^m e^{-\beta w_i \cdot F_i(G)}}{\mathcal{Z}_w}$$

where $\beta := (RT)^{-1}$, R is the Gas constant, T the temperature in Kelvin, and \mathcal{Z}_w denotes the **partition function**

$$\mathcal{Z}_w = \sum_{G \subseteq G_T} \prod_{i=1}^m e^{-\beta w_i \cdot F_i(G)}$$

559 We can forget about the β contribution as we can rewrite the weight $w'_i = \beta w_i$. The weights
 560 w_i are values chosen or tuned by us.

Tuning the weights is done by fixing a mean T^{F^*} (and T^F threshold) for each type of neighborhood. We can then tune the weight $w(F_i)$ to give more “importance” that will favor value around T^{F^*} . In practice, when a feature for a neighborhood varies greatly between instances, it means that this neighborhood is strongly relevant to distinguish the different matches. It gives us an incentive to modify its weight accordingly. To do so, instead of choosing weights manually, we solve the following problem:

$$\min_w \sum_{i=1}^m |\mathbb{E}[F_i \mid w] - F_i^*|$$

561 This problem is known to be convex. We used so convex optimization method. Further
 562 details about this problem, including the proof of convexity, are addressed in [15].

563 **A.2** Computation of the partition function using dynamic programming

564 **A.2.1** Definitions

565 First, we introduce the formal definition of the treewidth, we also depict what is a nice
 566 tree decomposition (NTD) as it allows a simpler search during the dynamic programming
 567 procedure. NTD implies no additional cost because an NTD has at most a size $n = |G_T|$

► **Definition 3** (Tree Decomposition (TD)). *Given a graph $G = (V, E)$, a tree decomposition of G is a tree T , whose nodes are bags $Y_1 \dots Y_t$ such that: (definition from Bodlander et al [3])*

- 570 1. $V \subset \bigcup_{i=1}^t Y_i$
- 571 2. $\forall (u, v) \in E, \exists i \in \llbracket 1, t \rrbracket, (u \in Y_i) \cap (v \in Y_i)$

572 3. $\forall u \in V, \{u | u \in Y_i\}$ is a subtree of T .

573 ► **Definition 4** (Nice Tree Decomposition). A tree decomposition T of $G = (V, E)$ is said
574 “nice” if each bags Y_i has one of the three following forms :

575 ■ *Introduce*: Node Y_i has exactly one child of index c in T and $Y_i = Y_c \cup \{v\}$

576 ■ *Forget*: Node Y_i has exactly one child of index c in T and $Y_c = Y_i \cup \{v\}$

577 ■ *Join*: Node Y_i has exactly two children of indices c_1 and c_2 in T and $Y_i = Y_{c_1} = Y_{c_2}$

► **Definition 5** (Treewidth). The treewidth ϕ of a graph G is defined as the biggest bag of the
“best” tree decomposition of G :

$$\phi = \min_{\text{tree dec. } T \text{ of } G} \max_{Y_i \in T} |Y_i| - 1$$

578 A.2.2 Dynamic programming solution

579 We now address the computation of the partition function [15] from 2 through a dynamic
580 programming procedure on the nice tree decomposition of G_T .

581 It is a bottom-up dynamic procedure (from leaves to the root) that relies on the following
582 different equations depending on the type of the node Y_i in the nice tree decomposition T .
583 We denote :

584 ■ The set of neighborhood thresholds: $F = (T^L, T^E, T^G)$

585 ■ M_i , **partial mapping** at node Y_i of T .

■ The **separator node** of Y_i , $\text{sep}(Y_i)$ chosen as the first element of the set S :

$$S = \{x \in Y_i \mid x \notin Y' \text{ with } Y' \text{ a children of } Y_i\}$$

586 We can point out that, with a nice tree decomposition, there exists only a unique choice
587 for this node and the set S is reduced to a singleton.

■ Given a partial mapping M_i , we introduce the following Boolean condition to map each
contribution to a single bag and avoid multiple computations of it:

$$C(u_1, u_2, Y_i, M_i) = (u_1 = \text{sep}(Y_i) \cap M_i(u_2) \neq \emptyset) \cup (u_2 = \text{sep}(Y_i) \cap M_i(u_1) \neq \emptyset)$$

From this we introduce $\Delta(\cdot)$ to denote the global contribution

$$\Delta(M'_i, G_T, Y_i, T^F) = \{d_{G_T}^F(u_1, u_2, M'_i) \mid C(u_1, u_2, Y_i, M'_i) \text{ is True}\}$$

588 We fill the dynamic programming table P that stores the partial computation of the partition
589 function with equations:

■ Forget Node Y_i with child Y' :

$$P[Y_i; M_i] = P[Y'; M_i]$$

■ Introduction Node, creating vertex $s := \text{sep}(Y_i) \in V_P$ having child Y' :

$$P[Y_i; M_i] = \sum_{v \in D(s | M_i)} P[Y'; M_i \cup (s \leftarrow v)] \times \prod_{\substack{T^F \in F \\ \delta \in \Delta(M_i \cup (s \leftarrow v), G_T, Y_i, T^F)}} e^{-\mu \cdot w(T^F) \cdot \delta}$$

where $D(v | M)$ denotes the set of admissible mappings for $v \in V_P$, consistent with prior
assignment M , such that:

$$D(v | M) := \begin{cases} V_T & \text{if } M = \emptyset \\ \bigcap_{\substack{u \in M \\ \text{s.t. } u \prec v}} \{x \in V_T \mid M(u) \prec x\} \bigcap_{\substack{u \in M \\ \text{s.t. } v \prec u}} \{x \in V_T \mid x \prec M(u)\} & \text{otherwise.} \end{cases}$$

■ Join Node :

$$P[Y_i; M_i] = \prod_{Y' \in \text{children}(Y_i)} P[Y'; M_i]$$

590 The backtracking step to retrieve the value of probability for each graph (and so the whole
 591 Boltzmann distribution as introduced in 2) uses the same type of equations but going from
 592 top to bottom: a number is drawn at each node to know if we have to add a value for current
 593 mapping, given the partial partition function computed at each step of the forward procedure.
 594 Both the forward and backward steps are currently known procedures that have been studied
 595 and automatized in a framework named **Infrared**. [33], which has the advantage to be quite
 596 permissive about the definition of the neighborhood cumulative differences.

597 A.3 Choice on technical parameters

598 For the choice of the radius R for creating slices of target graph G_T , given an extracted
 599 graph G from G_T centered in nucleotide c , we first defined $R(G) = \min_{j \in G} \text{GEO}(j, c)$. To be
 600 exhaustive with our search, we must ensure that every G from G_T is extracted with a radius
 601 at least equals to $R(G_P) + D_{\text{gap}}$ as it ensures that we have enough "space" to make G_P fit
 602 in G even if some gaps occur. It is due to these gaps that we need to add D_{gap} in R . It
 603 embodies the specific case where the gap would have increased the length of the motif to
 604 search in G_T in a single direction by putting gaps one after the other. Due to the rarity of
 605 this case, we choose, in the tests, to use a smaller radius equal to $R(G_P) + \frac{D_{\text{gap}}}{4}$. The only
 606 taken risk here is to miss some patterns, but it is more convenient to favor time convergence
 607 as the pathological case on gaps evoked above is not one that we would like to target.

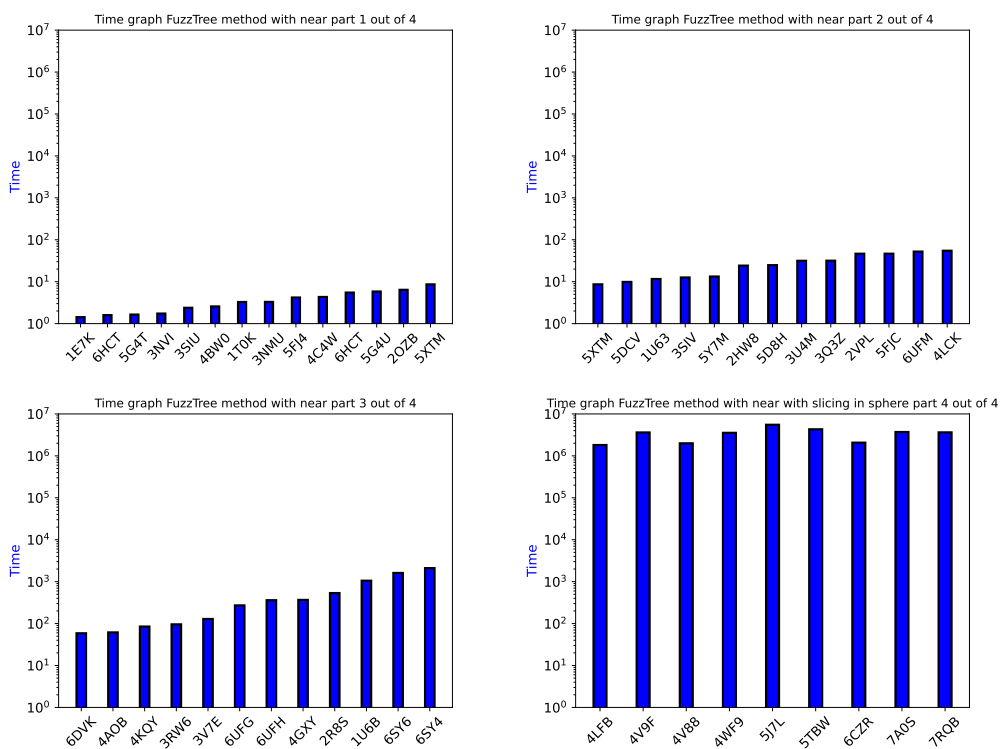
608 We also choose to use a timeout equal to 2000 seconds for the convergence of our algorithm
 609 on each extracted graph. Here again, the only risk is to miss some additional patterns.
 610 Nonetheless, all these limitations only mean that our current results can probably be slightly
 611 better regarding expressiveness, which means that somebody with more computational
 612 resources could use this tool and wait for even better performances.

613 A.4 Time results on Narval and Beluga clusters for FuzzTree

614 For this paper, computations were done on the Narval cluster and the Beluga cluster of the
 615 Digital Research Alliance of Canada. Each used node on Narval is made of 64 cores with 2
 616 CPUs AMD Rome 7532 @ 2.40 GHz. Each used node on Beluga is made of 40 cores with 2
 617 CPUs Intel Gold 6148 Skylake @ 2.4 GHz. Multiprocessing was used simply by separating
 618 the computations by chains of the same RNA and next, when relevant, by slices identified in
 619 these RNA chains.

620 Some time results for computation of the FuzzTree method, by requesting one motif
 621 on each RNA chain where Kink-Turns are known, are available in Fig. 12. The time
 622 of computation is large but it is something expected with the XP theoretical complexity.
 623 However, one can notice that in practice the treewidth of the selected pattern is equal
 624 to 2 which allows a complexity in $O(n^3)$. No true time discrepancy appears between the
 625 computation without near edges and the one with. On large graphs, due to the slicing, the
 626 time of computation is reduced, but such reduction is not perfect as slicing computation is
 627 still quite redundant: multiple graphs cover sometimes the same portion of the Kink-Turn.

12:22 Exploring the natural fuzziness of RNA non-canonical geometries



■ **Figure 12** Time graph of the FuzzTree method on each group of studied RNA chains.

On the Beluga cluster, computations were done on 1 processor for small RNAs (less than 500 nucleotides, which corresponds to the three first graphs) and on 40 processors for large RNAs (more than 500 nucleotides, which corresponds to the fourth graph). In that case, the depicted time is the sum of each time consumed for each processor.