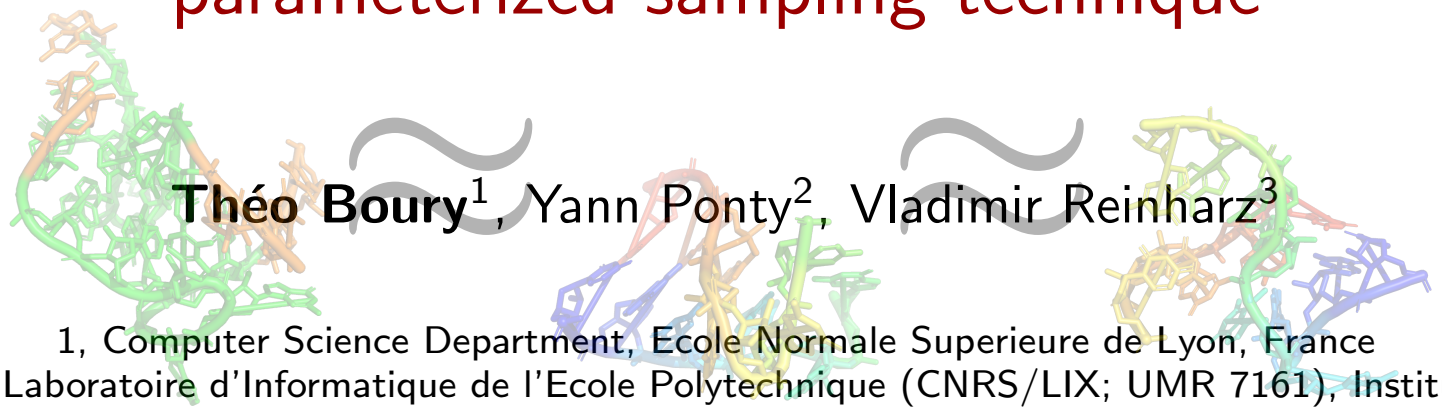
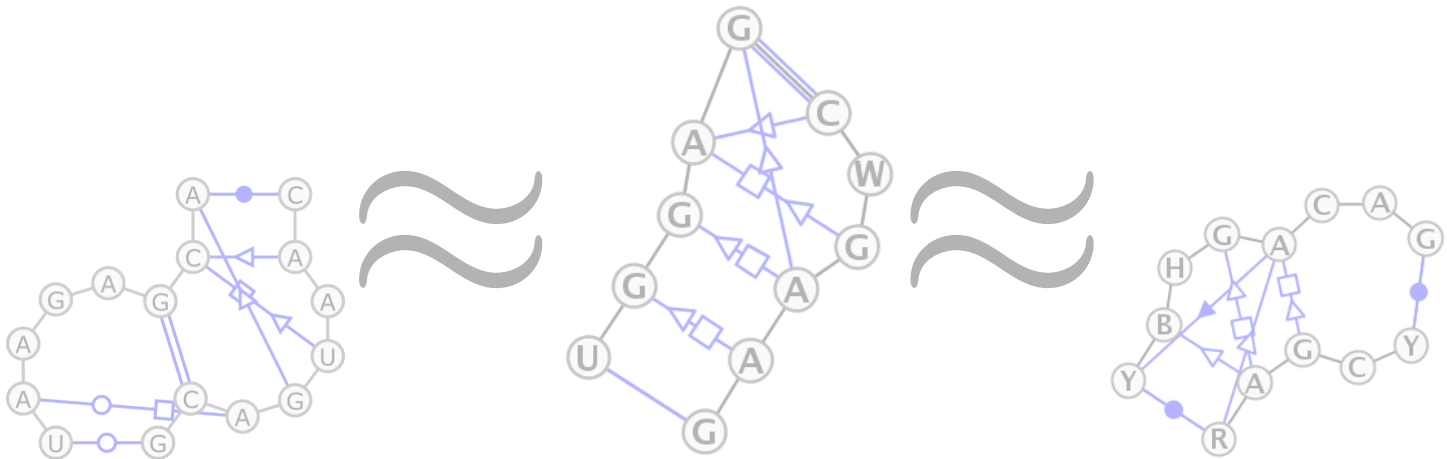
Automatic exploration of the natural variability of RNA non-canonical geometric patterns with a parameterized sampling technique



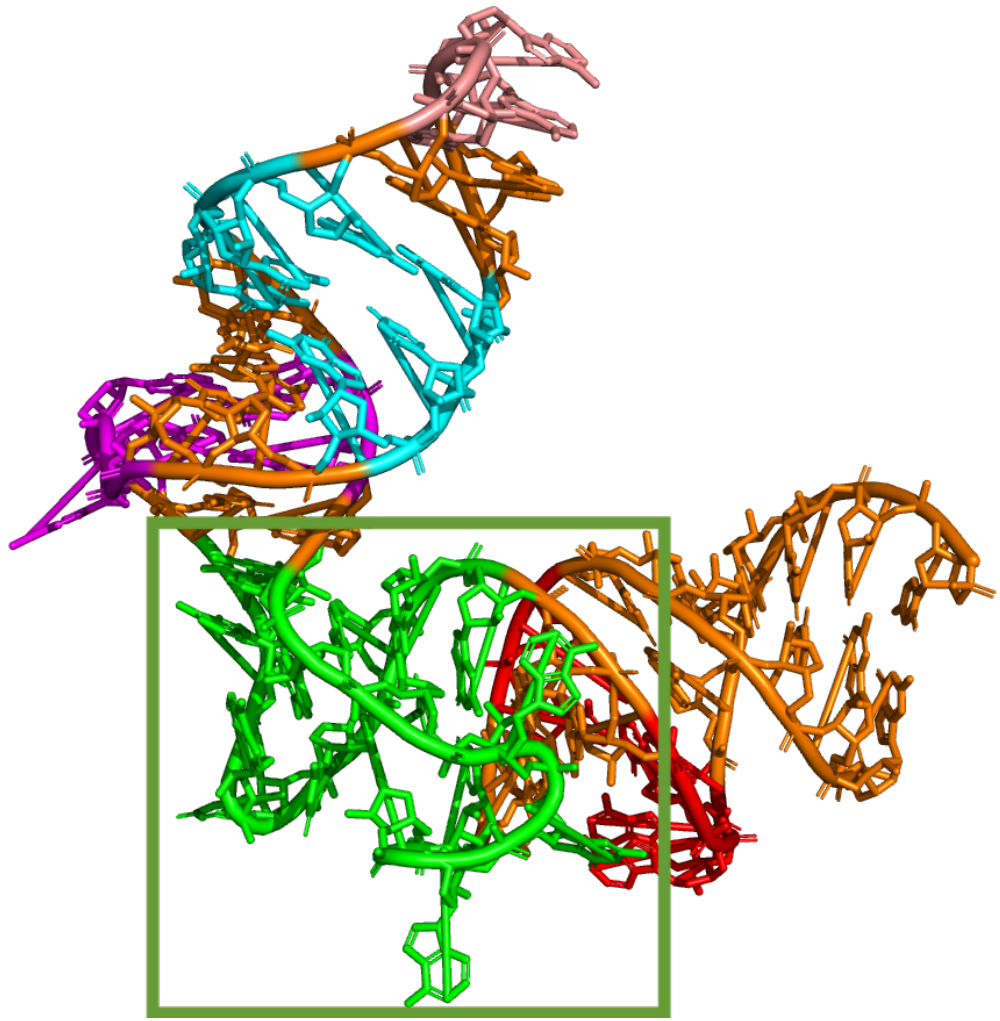
Théo Boury[1], Yann Ponty[2], Vladimir Reinharz[3]

1, Computer Science Department, Ecole Normale Superieure de Lyon, France
2, Laboratoire d'Informatique de l'Ecole Polytechnique (CNRS/LIX; UMR 7161), Institut Polytechnique de Paris, France
3, Department of Computer Science, Université du Québec à Montréal, Canada

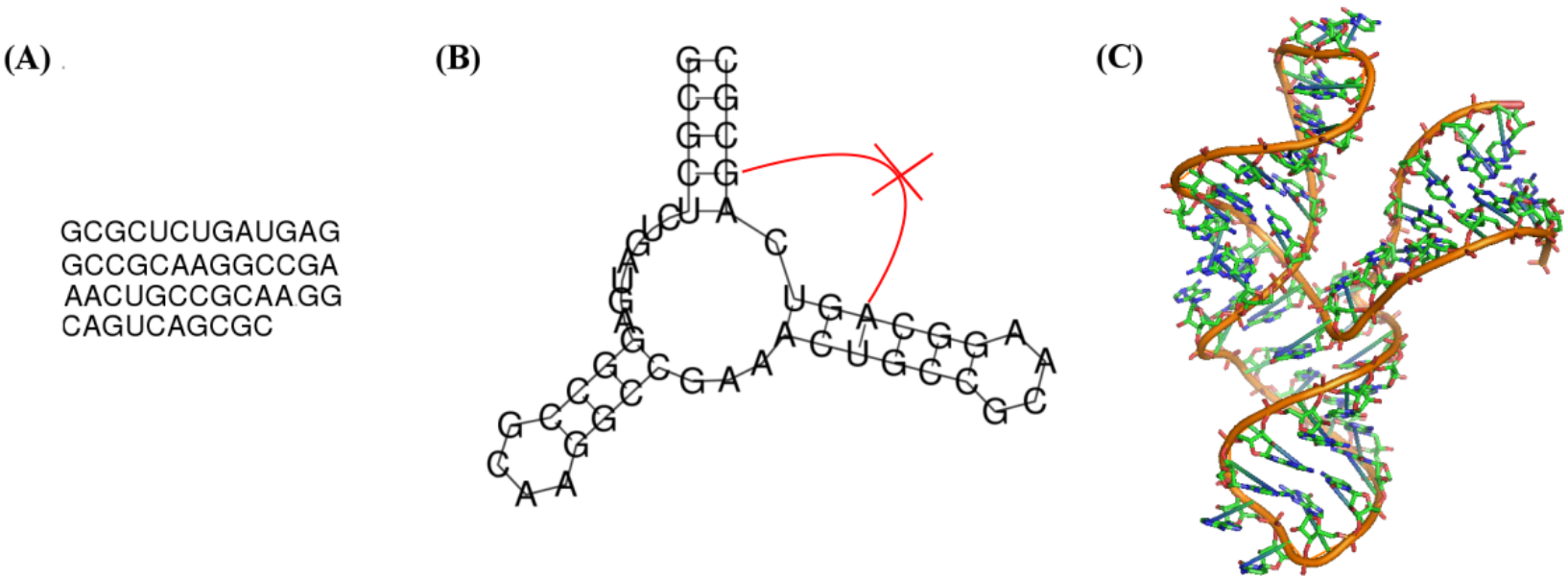


The 3D RNA structure

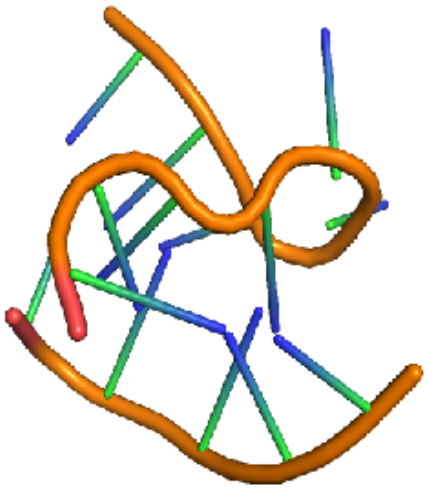
The 3D RNA structure



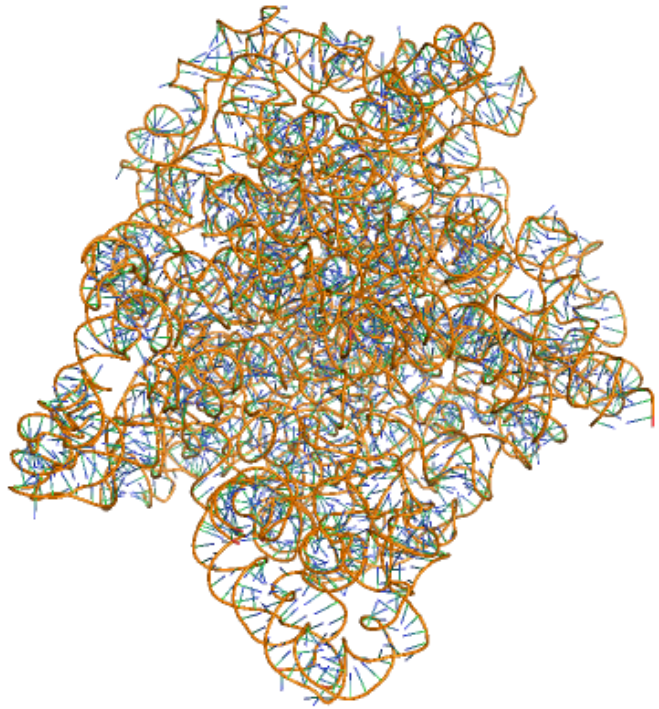
Different level of abstraction for RNA



Where is Waldo?

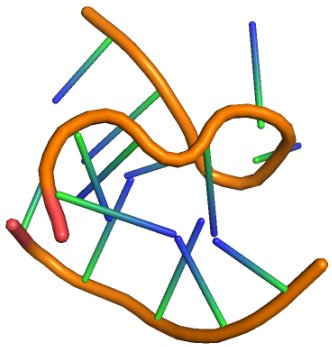


Motif

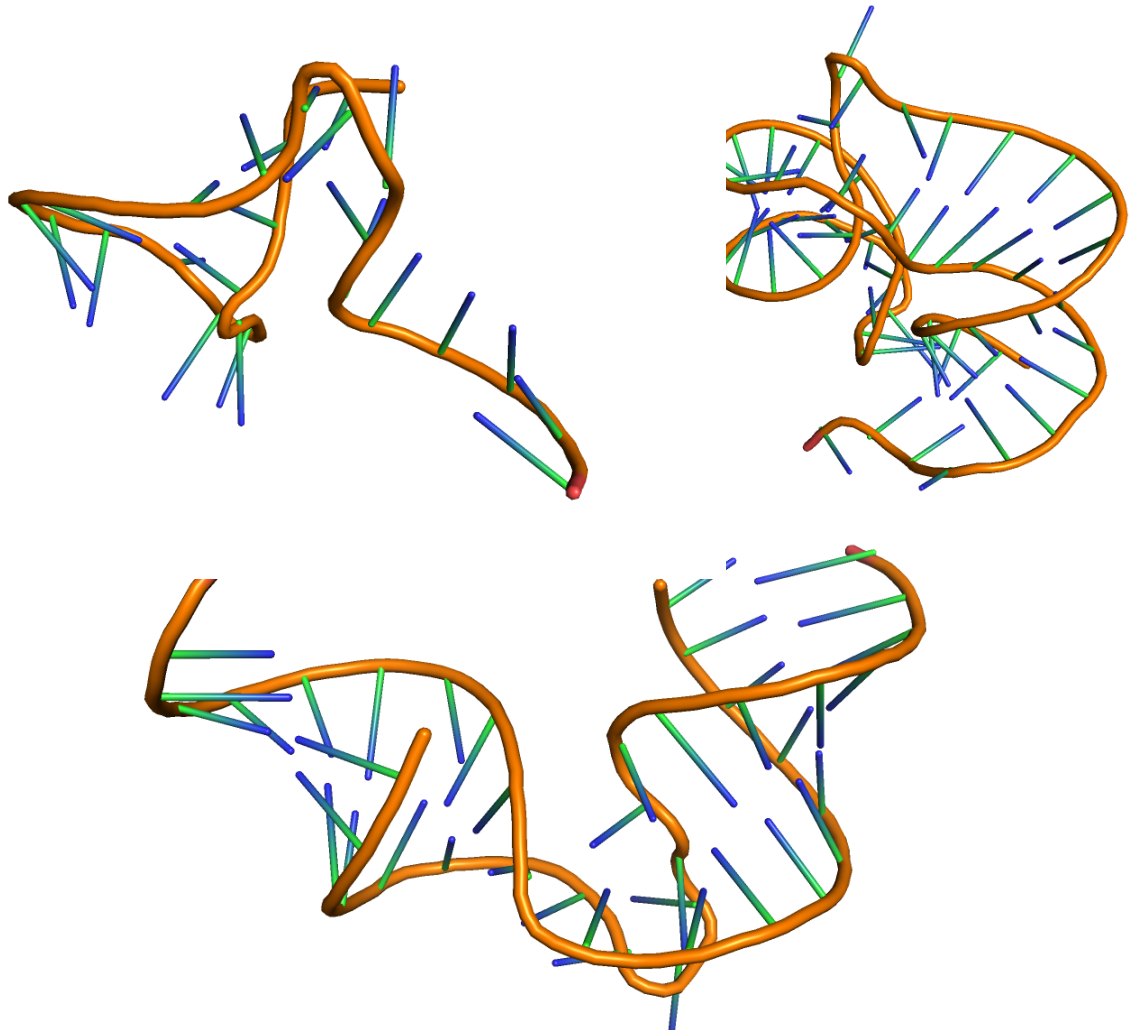


RNA 4V9F

Where is Waldo?

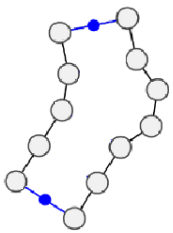


Motif

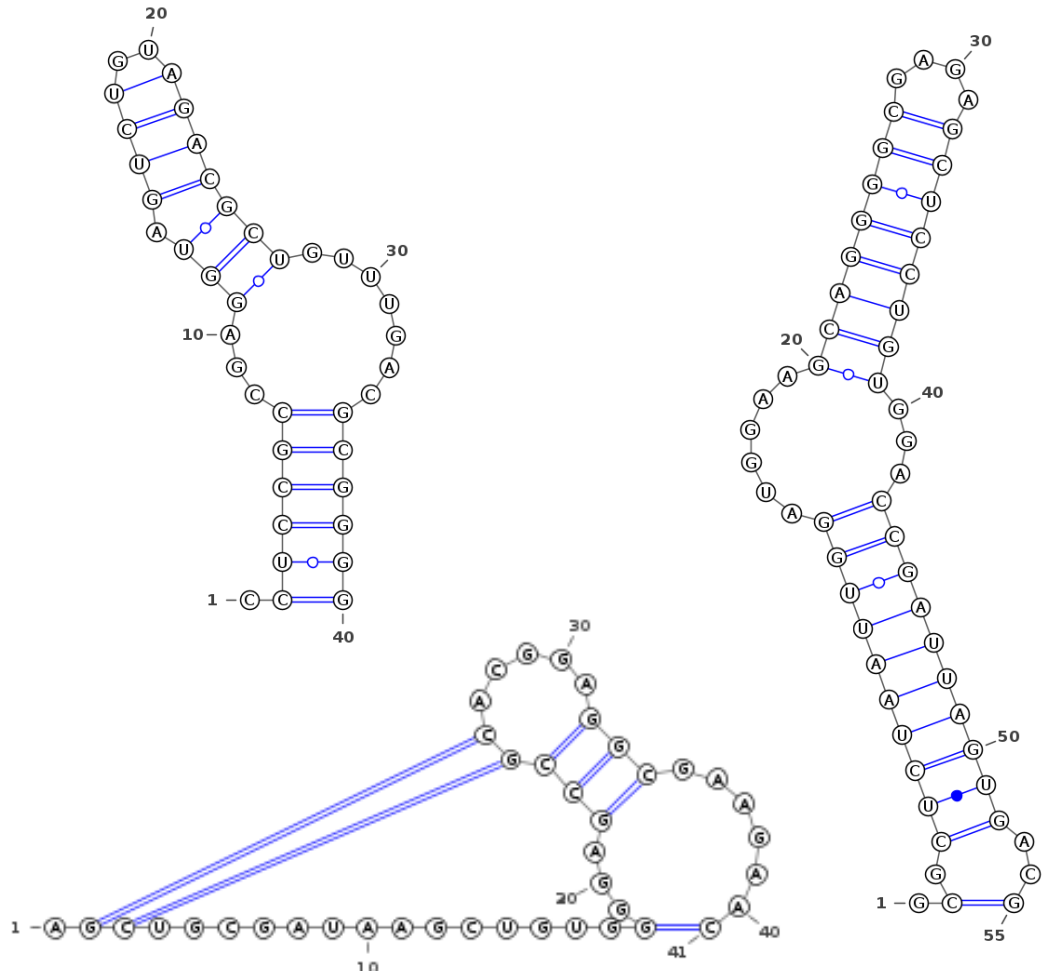


Subparts of RNA 4V9F

Where is Waldo?

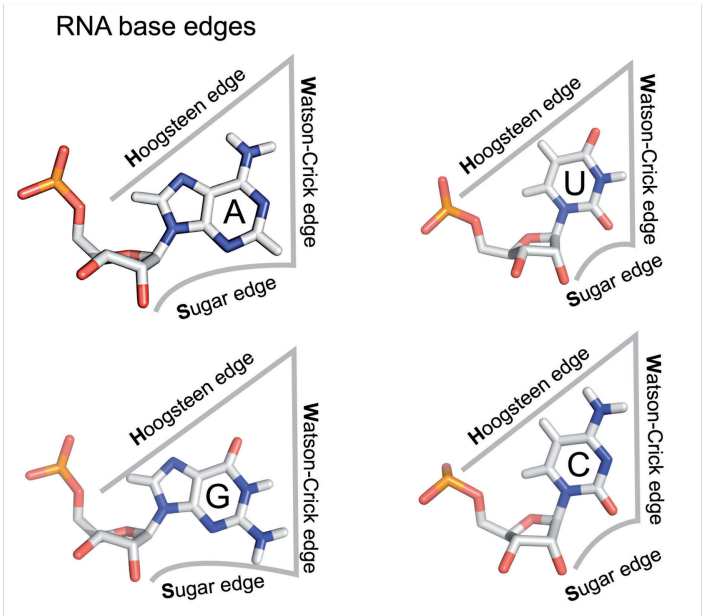


Motif



Subparts of RNA 4V9F

Non canonical annotations (Leontis-Westhof) to the rescue!



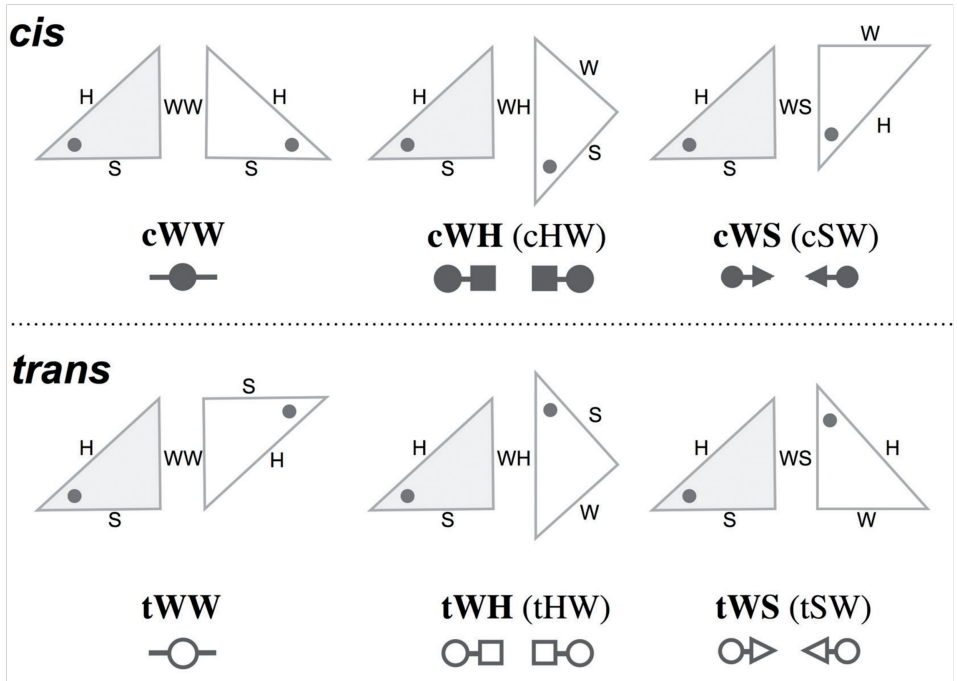
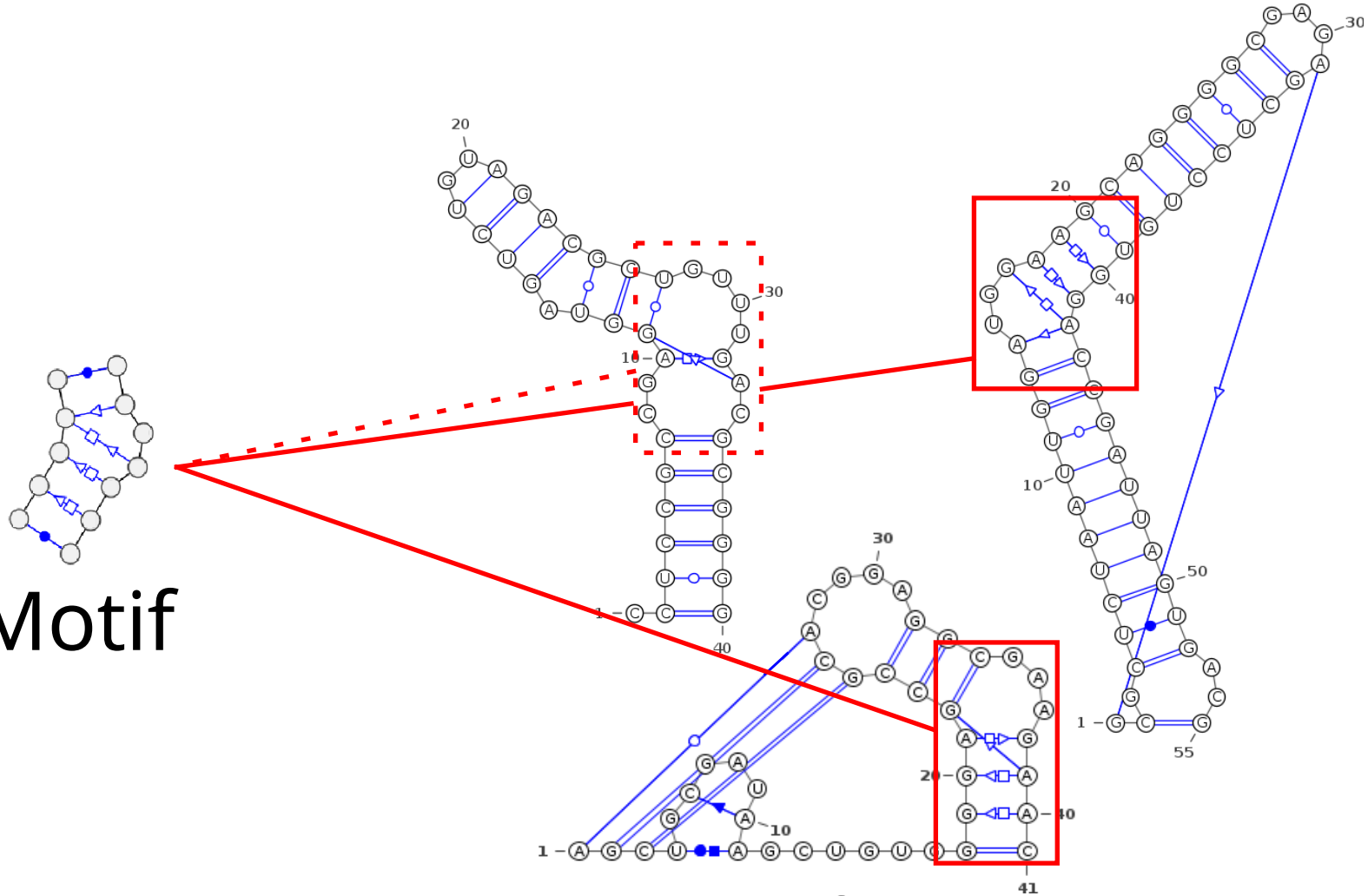
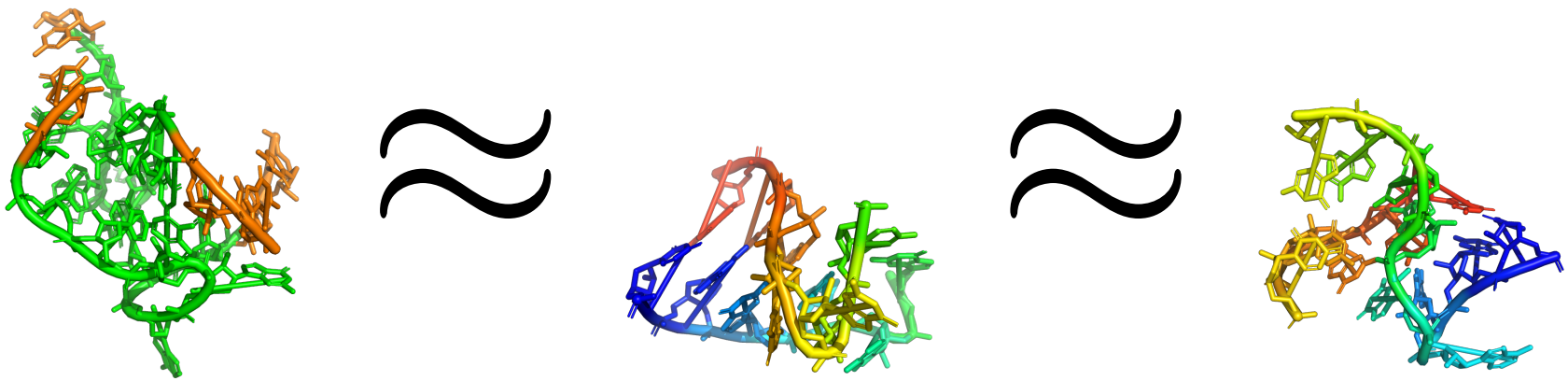


Figure adapted from Almakarem et al, 2011

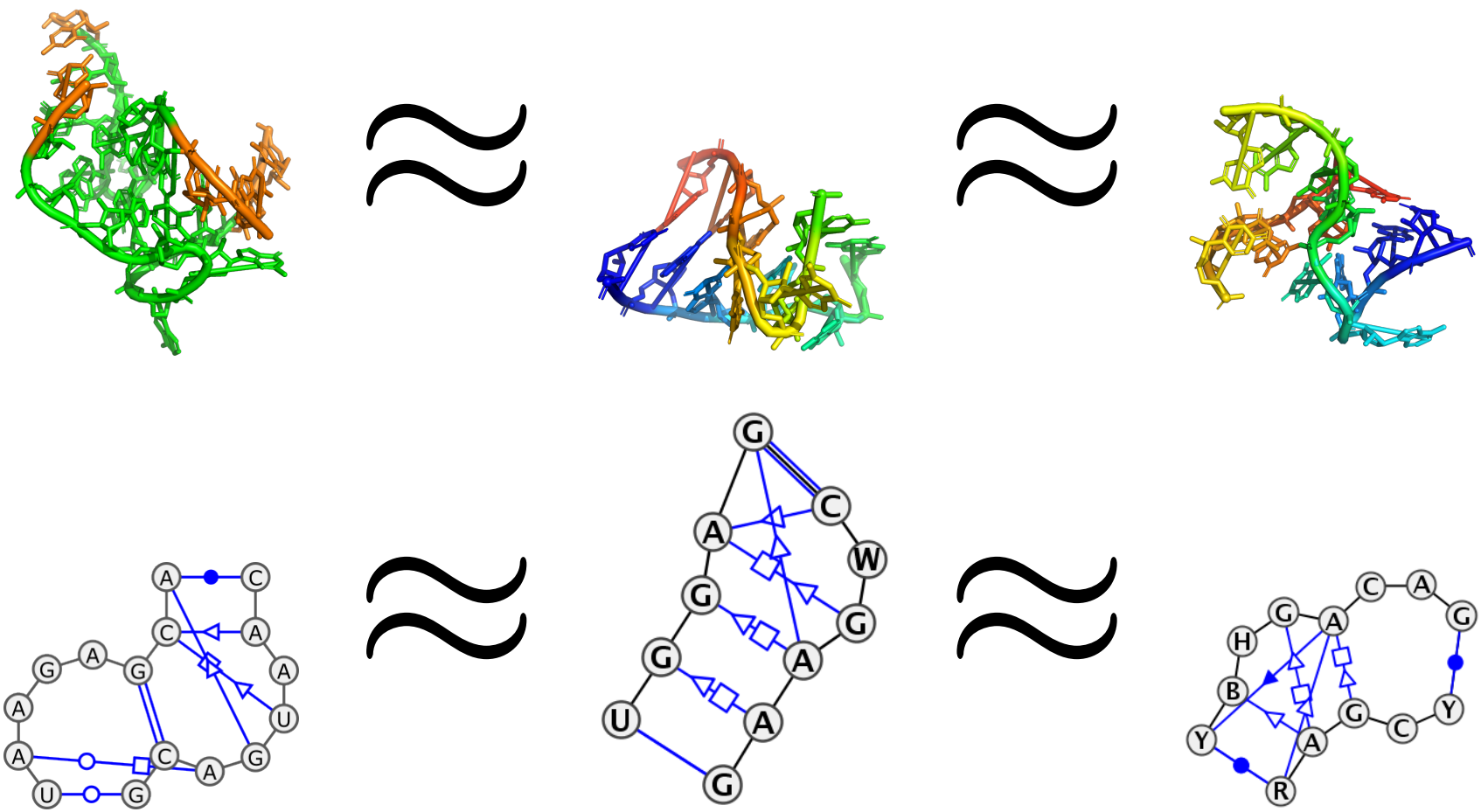
Where is Waldo (again)?



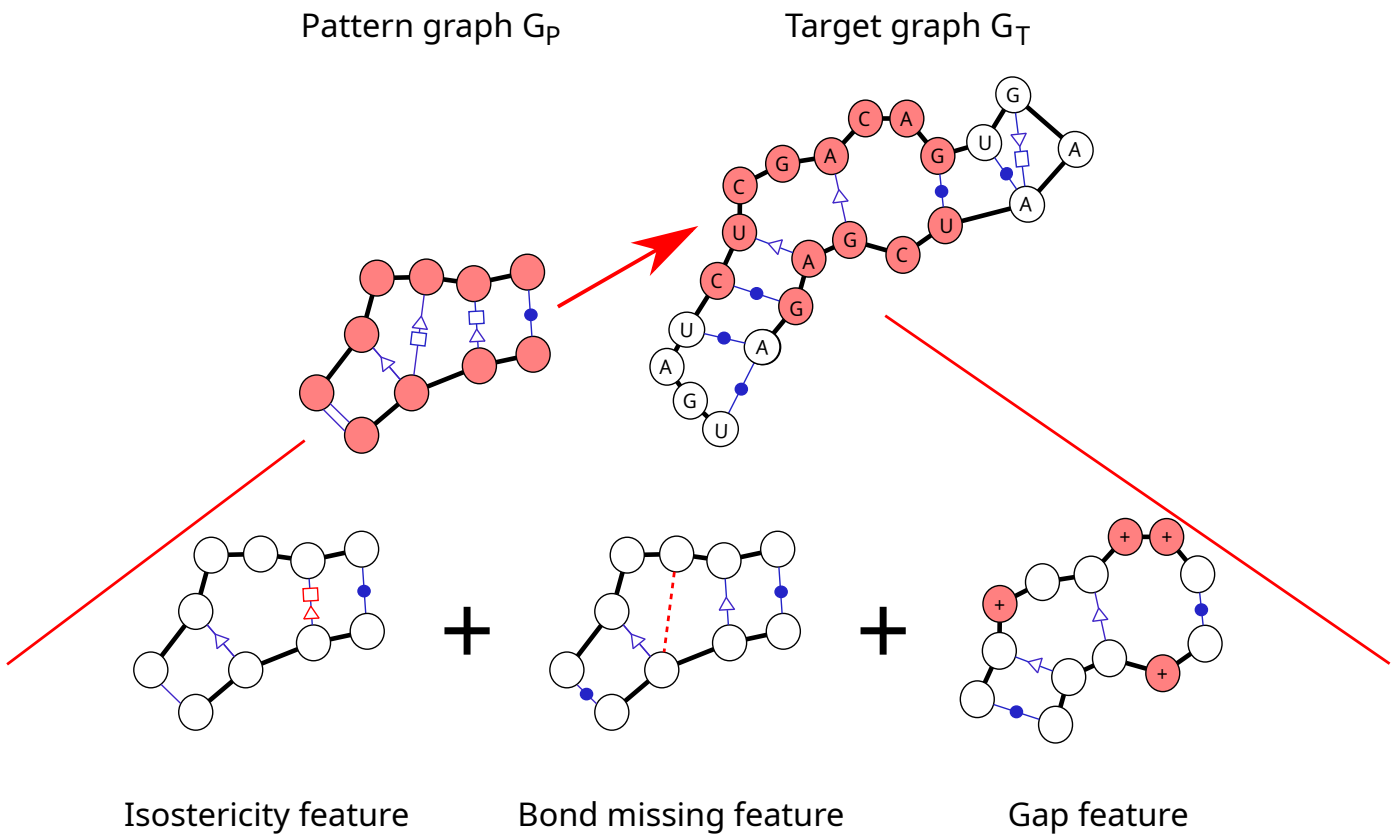
3D homology...



...Is not always obvious even with the non-canonical structure



Achieved result



New method : FuzzTree

- Sample RNA subgraphs in a neighborhood of G_P.
- Used neighborhoods: isostericity, missing bonds and gaps.
- Complexity: XP in G_P treewidth.
- Other state-of-the-art methods: only exact matches.

Problem formalism

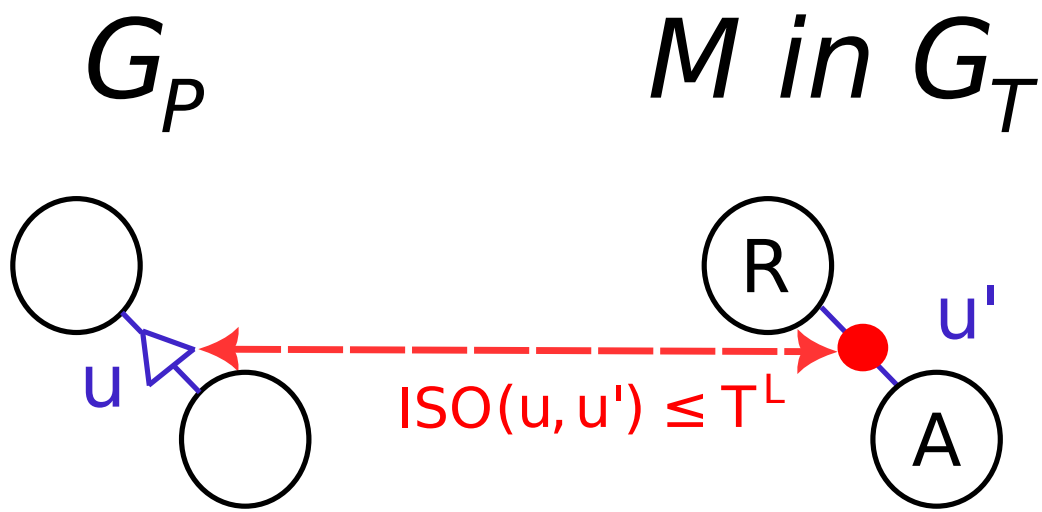
Input: Pattern graph $G_P = \left(V_P, E_P = B_P \sqcup \overline{B}_P\right)$ ($\prec$ −Hamiltonian) , target graph $G_T = \left(V_T, E_T = B_T \sqcup \overline{B}_T\right)$ and neighborhood thresholds (T^L, T^E, T^G, D_{edge}, D_{gap})

Output: Mapping $M : V_P \to V_T$ such that:

1. $\forall (u, v) \in {V_P}^2, u \prec v \Rightarrow M(u) \prec M(v)$ **(monotonicity)**
2. $\sum_{(u,v)\in\overline{B}_P} \text{ISO}\left(L\left(u, v\right), L\left(M(u), M(v)\right)\right) \leqslant T^L$ **(label compatibility)**
3. $\sum_{(u,v)\in\overline{B}_P} 1 - \mathbb{1}_{(M(u),M(v))\in\overline{B}_T} \leqslant T^E$ **(few missing edges)**
4. $\forall (u, v) \in \overline{B}_P, (M(u), M(v)) \notin \overline{B}_T, \text{GEO}\left(M(u), M(v)\right) \leqslant D_{\text{edge}}$ **(edge distance limit)**
5. $\sum_{(p_0,\ldots,p_k)\in P, k\geqslant 3} \text{GEO}(p_0, p_k) \leqslant T^G$ **(path size limitation)**
6. $\forall (u, v) \in B_P, \exists (p_0, p_1, p_2, ..., p_k) \in P$ such that **(no missing backbone path)**
   - $p_0 = M\left(u\right), p_k = M\left(v\right)$ **(*)**
   - $\text{GEO}\left(p_0, p_k\right) \leqslant D_{\text{gap}}$ **(**)**

or $\varnothing$ if no such mapping exists.

The label compatibility feature d^L



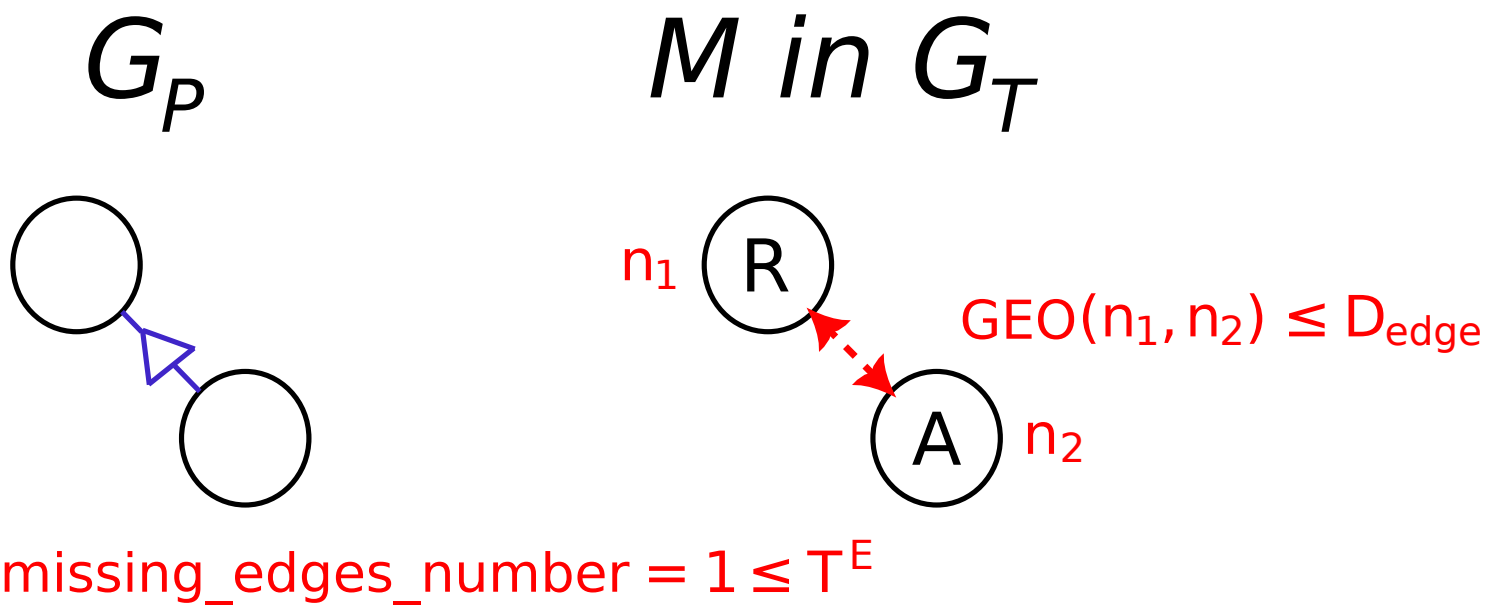
Definition 2.1 (Label difference d^L):

$$d^L_{G_T}(u, v, M) = ISO\left(\text{Label}(u, v), \text{Label}(M(u), M(v))\right)$$

- Isostericity ISO[1] compares both the 12 canonical and non-canonical base pairing families.

[1] Stombaugh et al, 2009, Nucleic Acids Research

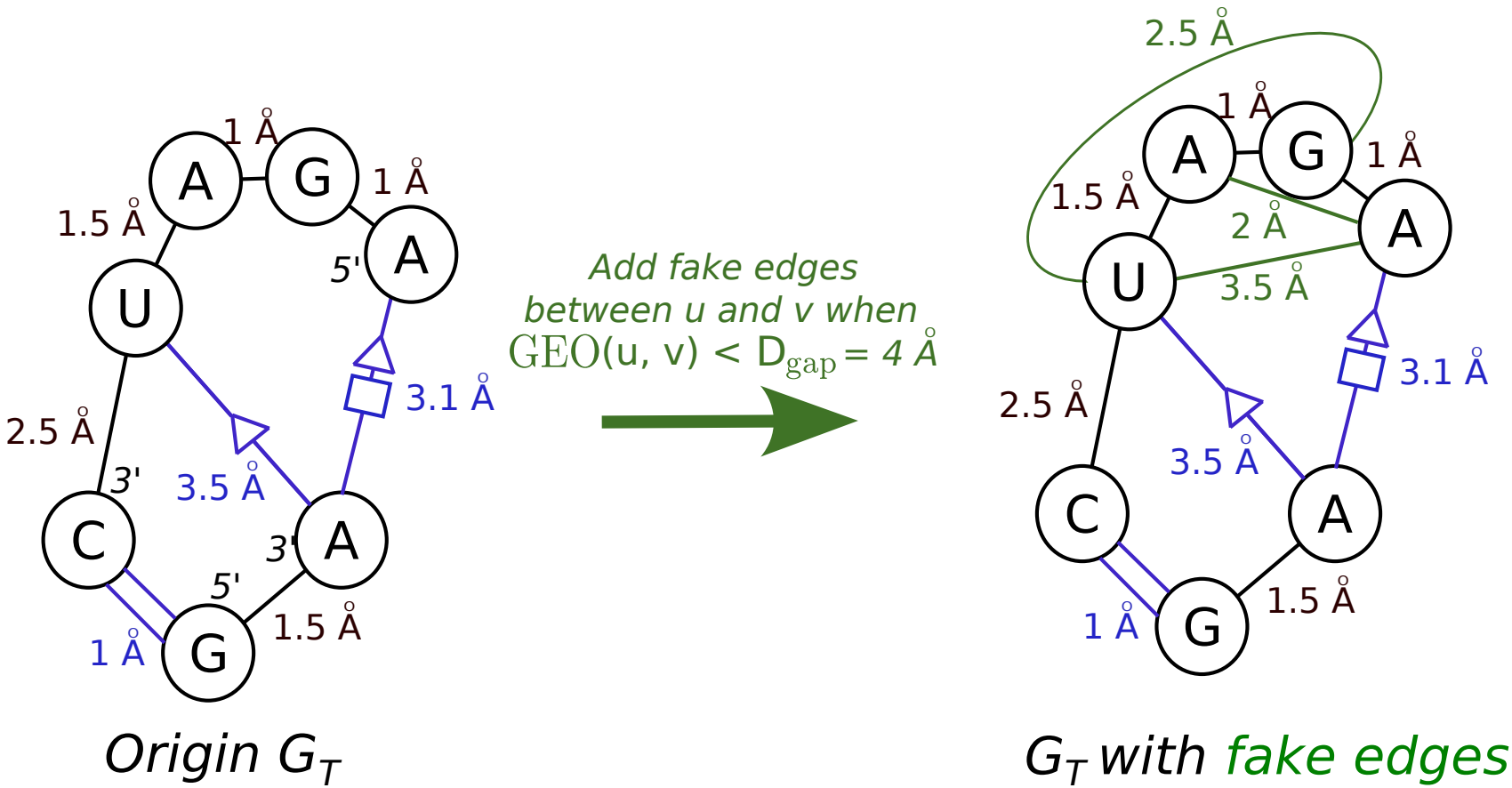
The few missing edge feature d^E



Definition 2.2 (Edge difference d^E):

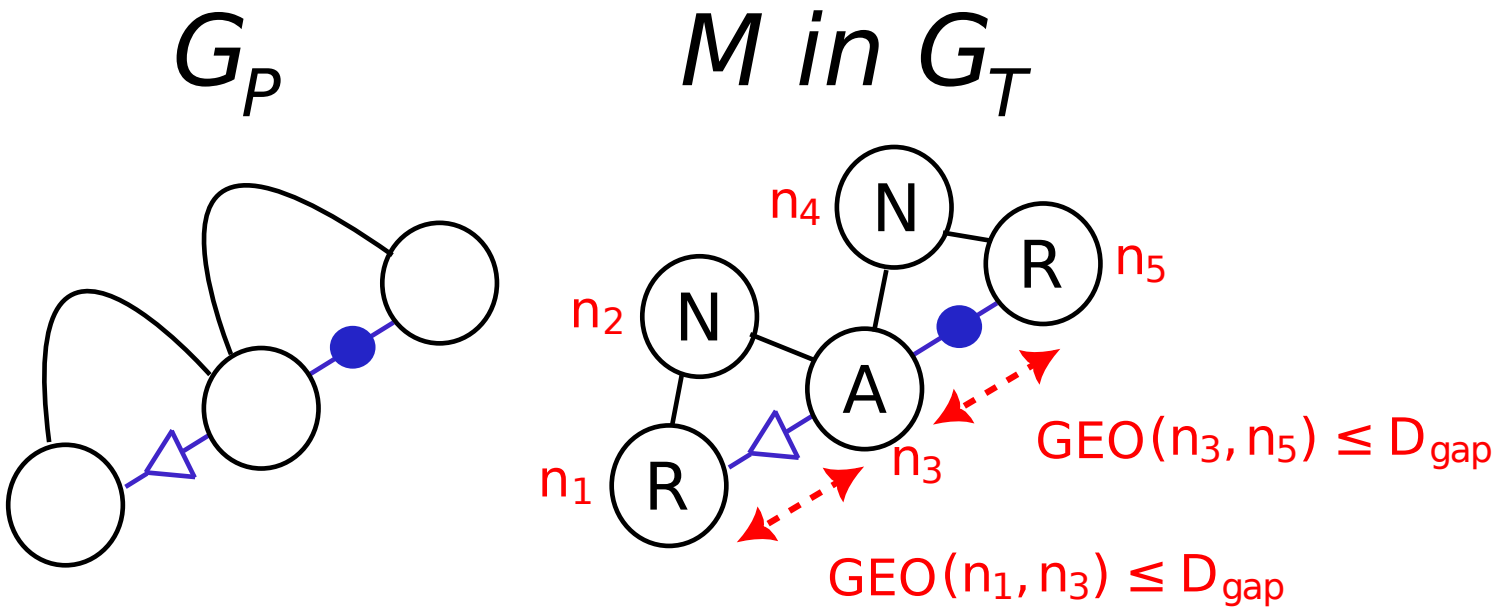
$$d^E_{G_T}(u, v, M) = \begin{cases} 0 & \text{if } (u,v) \in B_P \cap (M(u), M(v)) \in B_T \\ & \text{or } (u,v) \in \overline{B}_P \cap (M(u), M(v)) \in \overline{B}_T \\ 1 & \text{if } (u,v) \in \overline{B}_P \cap (M(u), M(v)) \notin \overline{B}_T \\ & \text{and } \mathrm{GEO}(M(u), M(v)) \leqslant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

Gap false edges creation



- Fake edges addition only when distances between consecutive backbones distances are small.

The gap feature d^G



Definition 2.3 (Gap difference d^G):

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

Corresponding NP-complete Problem

Our problem specializes in Hamiltonian Subgraph Isomorphism Problem, known to be NP-complete:

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

Output: Mapping $M : V_P \to V_T$ such that

- $\forall (u, v) \in {V_P}^2, u \prec v \Rightarrow M(u) \prec M(v)$ **(monotonicity)**
- $\forall (u, v) \in E_P, (M(u), M(v)) \in E_T \Rightarrow L((u, v)) = L((M(u), M(v)))$ **(label comp.)**
- $\forall (u, v) \in E_P, (M(u), M(v)) \in E_T$ **(no missing edge)**

or $\varnothing$ if no such mapping exists.

Accounting the features to get a subset of solutions

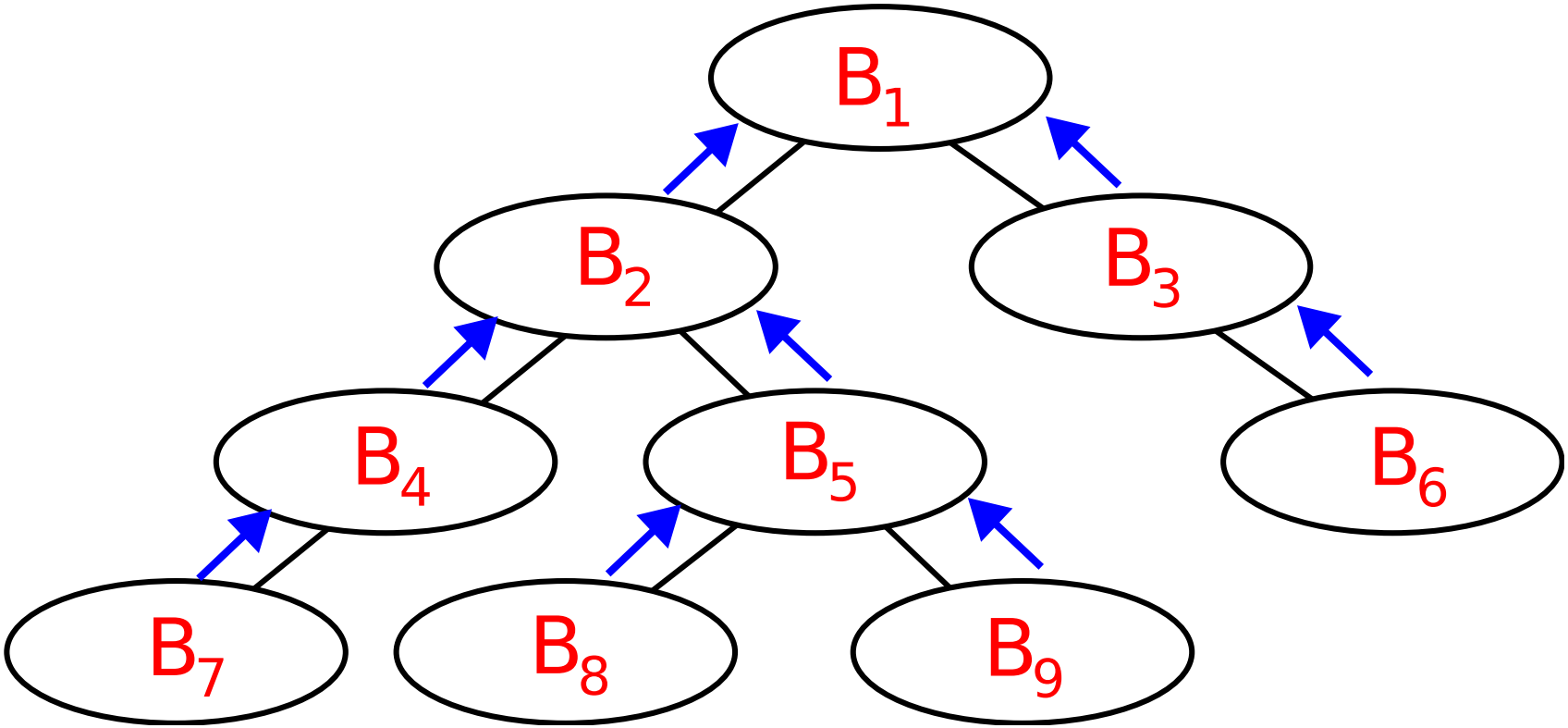
Features of a mapping M are taken into account through an additive (pseudo-)energy function:

$$E(M) = \sum_{(u,v)\in E_P} w_L \times d^L(u,v,M) + w_E \times d^E(u,v,M) + w_G \times d^G(u,v,M)$$

Where w_L, w_E, w_G are real positive valued weights.

- Exact mapping corresponds to $E(M) = 0$.
- $E(M) \neq 0$ and $E(M) << \infty$ corresponds to fuzzy matches.

Computation of dynamic programming procedures on trees

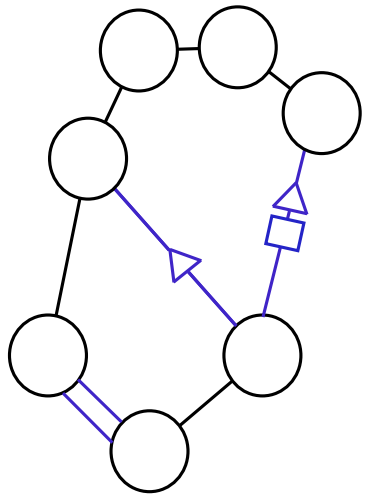


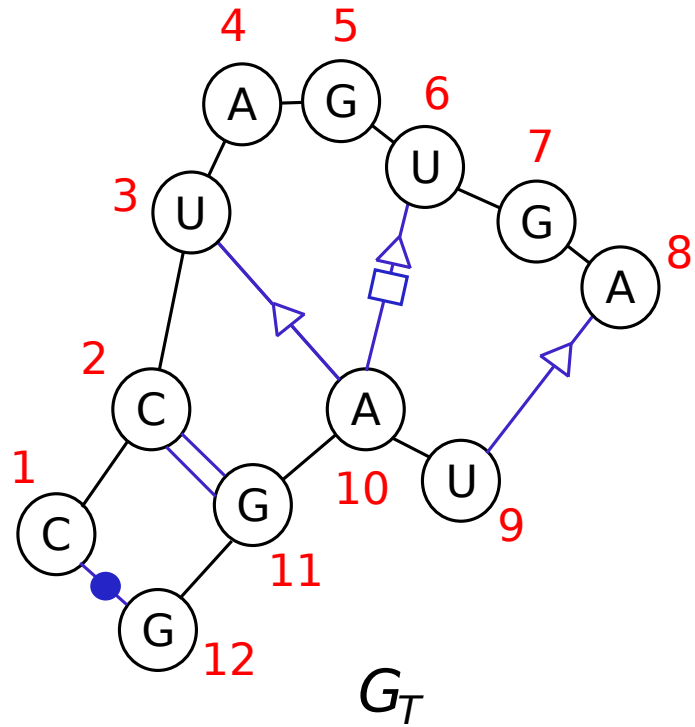
- Dynamic programming on trees, in particular on tree-decomposed instances, is automatized with the Infrared [2] framework.

[2] Hua-Ting et al, 2022, RNA Folding - Methods and Protocols

Decompose our instance into tree

Origin G_P



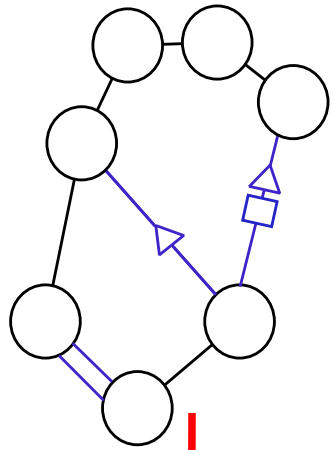


G_T

— *Backbone from 5' to 3'*
— *Labeled base pair*

Decompose our instance into tree

Origin G_P

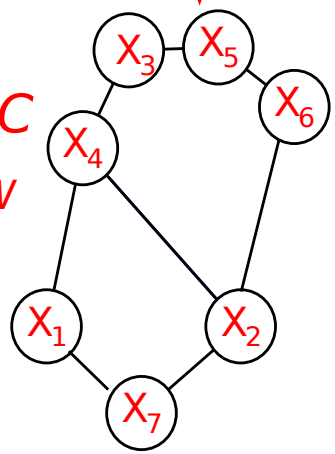


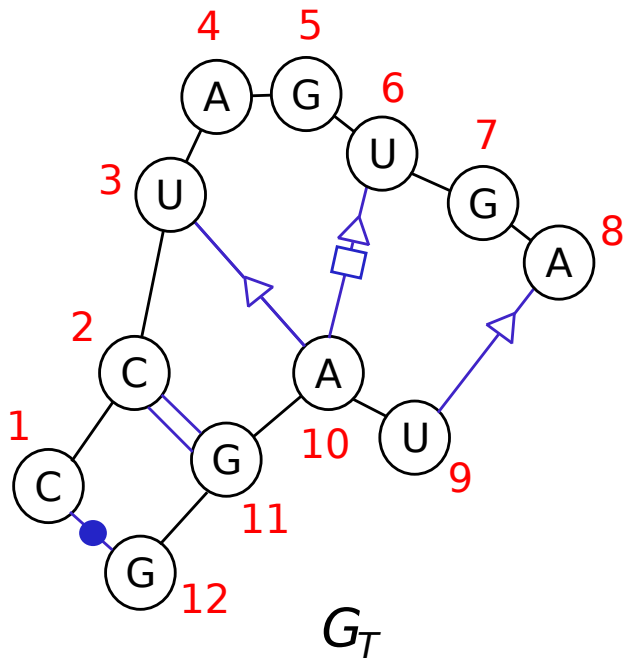
Constraints graph C

label(X_1, X_7) = cWW

label(X_2, X_4) = tSS

label(X_2, X_6) = tHS





G_T

— *Backbone from 5' to 3'*

— *Labeled base pair*

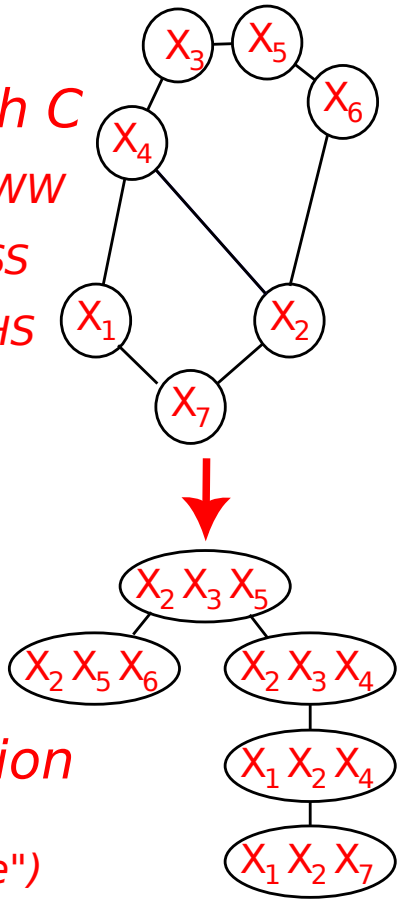
Decompose our instance into tree

Constraints graph C

label(X_1, X_7) = cWW

label(X_2, X_4) = tSS

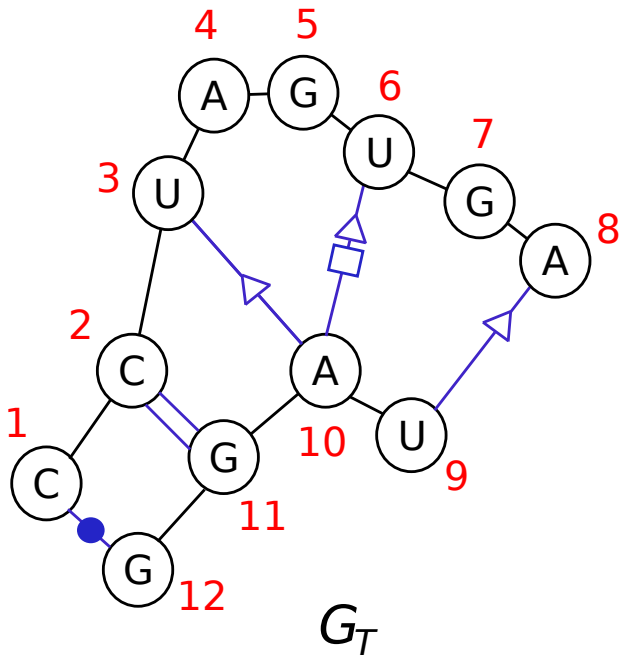
label(X_2, X_6) = tHS



Tree Decomposition

T(C)

(In reality, T(C) is``nice")



G_T

— *Backbone from 5' to 3'*

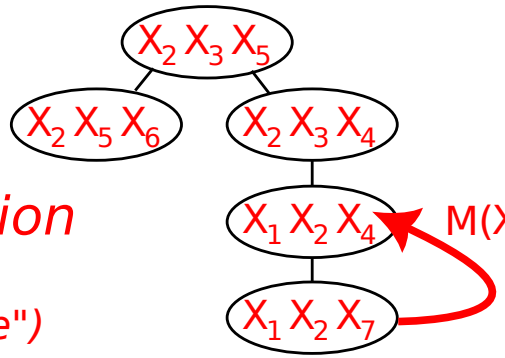
— *Labeled base pair*

Decompose our instance into tree

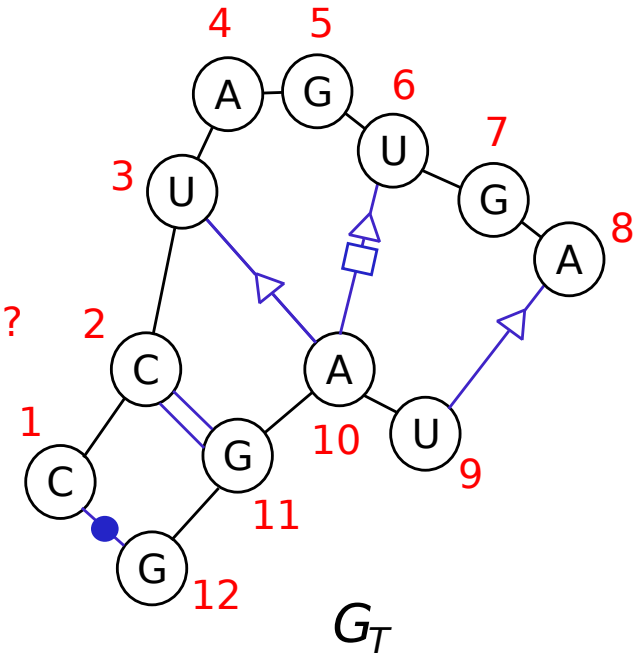
Tree Decomposition

T(C)

(In reality, T(C) is ``nice")



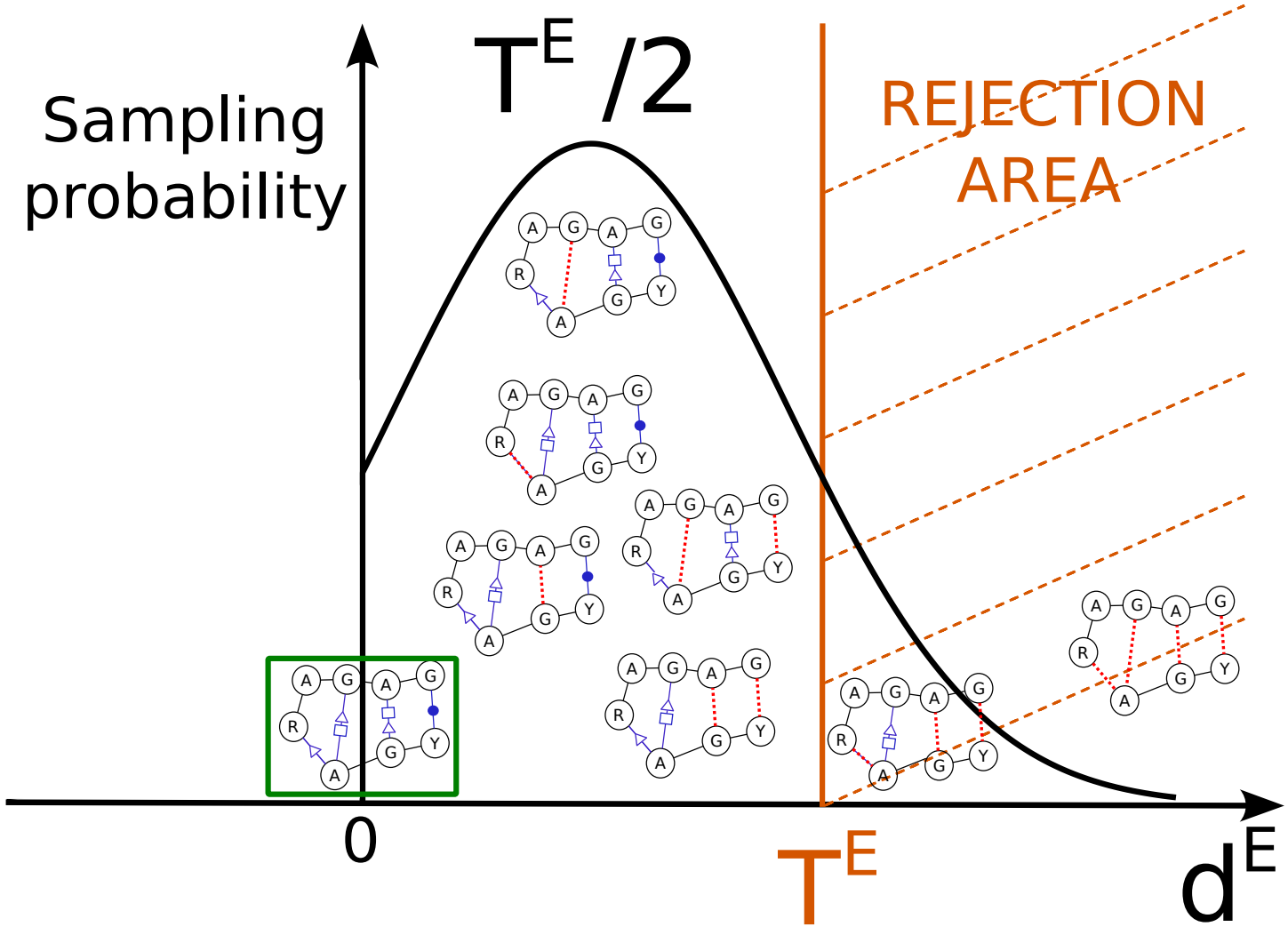
$M(X_1) = 2$
$M(X_2) = 10$
$M(X_7) = 11$



— *Backbone from 5' to 3'*
— *Labeled base pair*

- Complexity: $O\left(kn^{\phi+1}\right)$
- n and k, respective number of nodes in G_T and G_P and ϕ, the treewidth of G_P.

Sampling into a Boltzmann distribution with Infrared

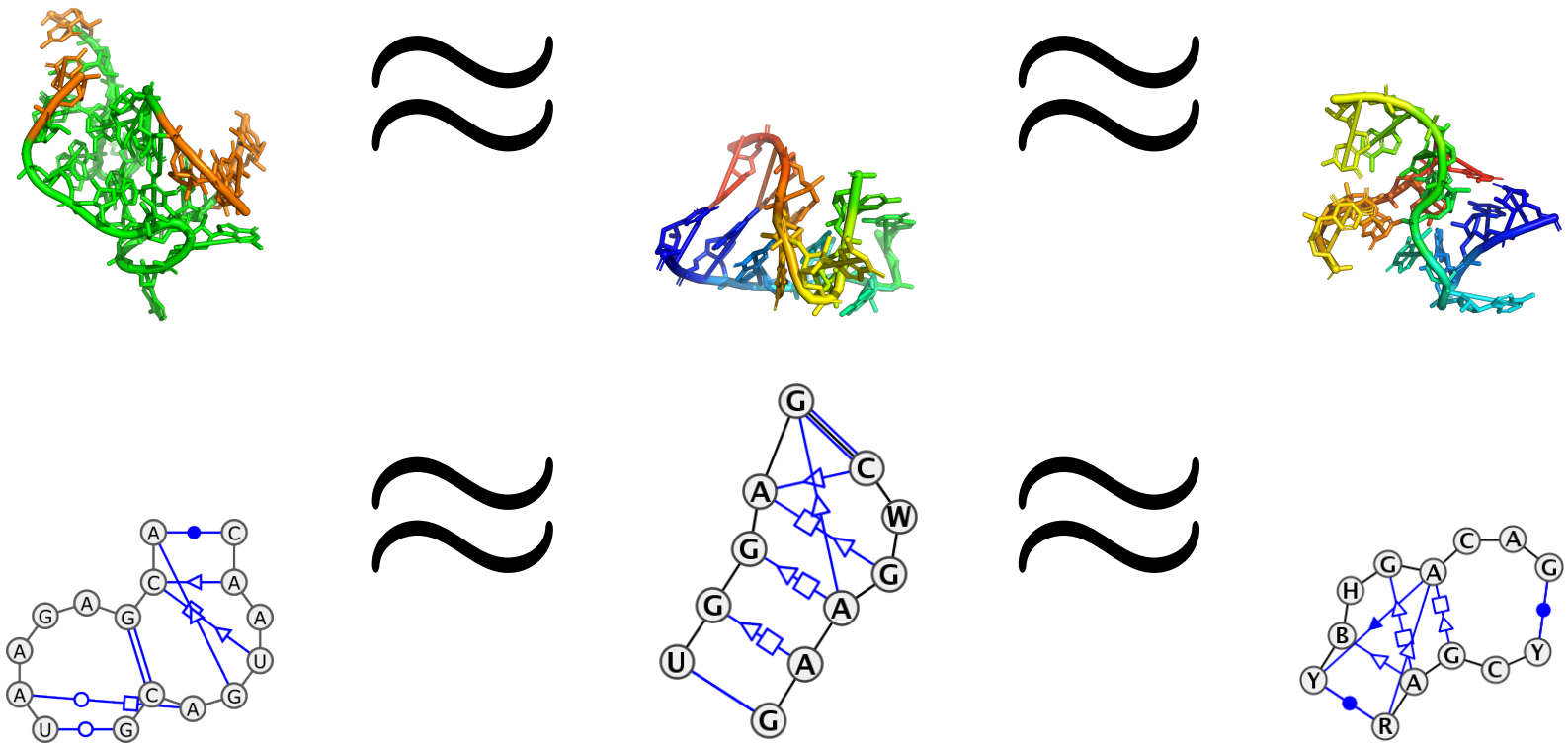


- We sample (given a pseudo-energy) instead of simply searching an optimal.
- Complexity: (sampling only) $O(knt)$, with t, number of samples.

Validation dataset

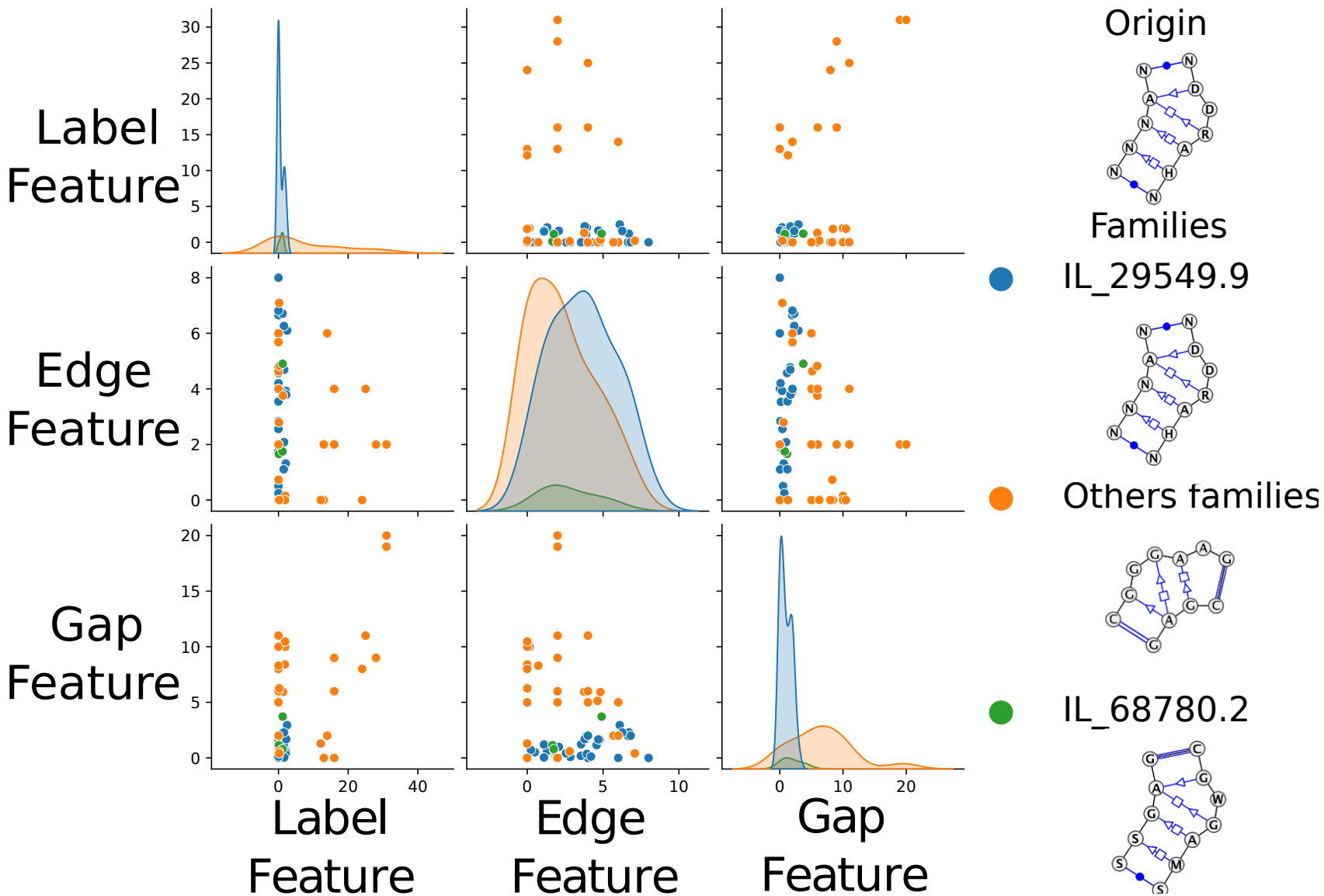
The Kink-Turn family dataset:

- A biological family that contains 72 known motifs over more than 25 different RNAs.
- Kink-Turns are clustered in 18 different families according to atomic cristallography. [3]



[3] Petrov et al, 2013, RNA

Kink-Turn Cartography

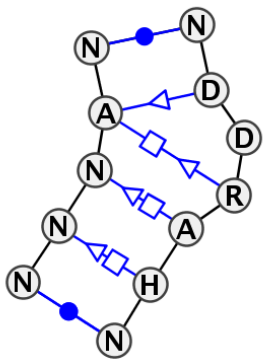


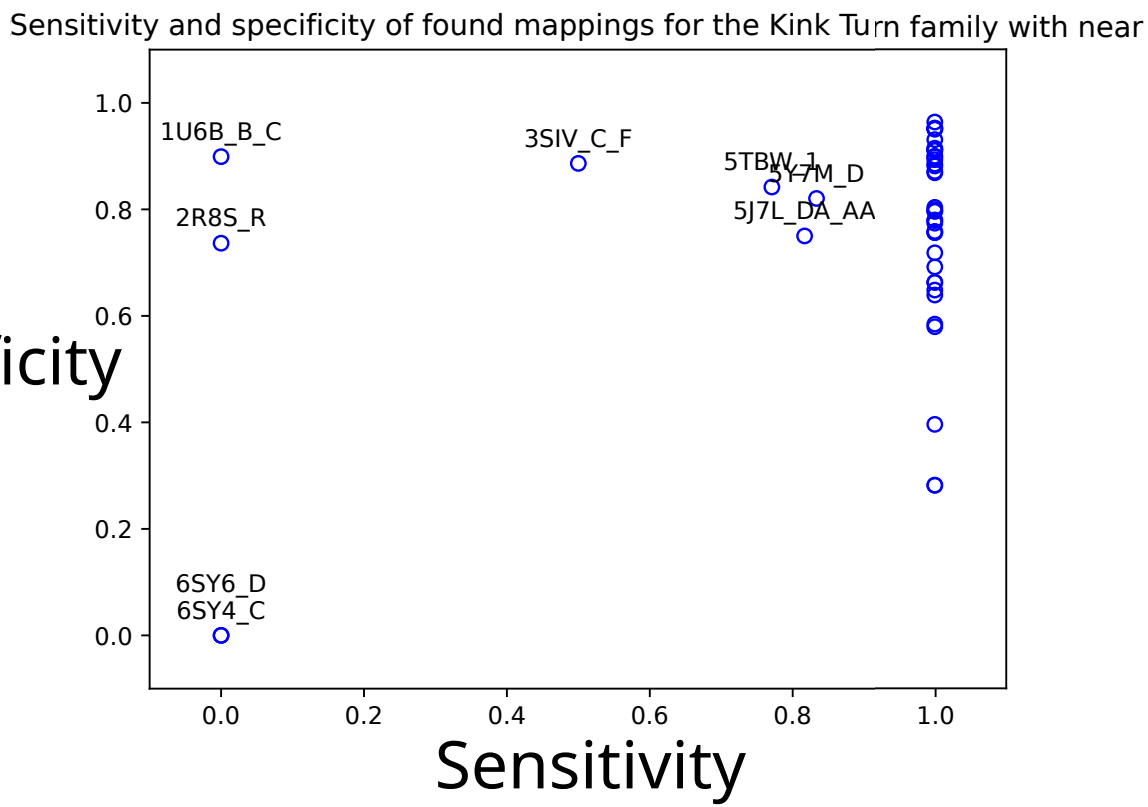
Retrieve the Kink-Turn family by requesting a single motif

We requested motif IL_5TBW_059 inside all RNA containing Kink-Turns.

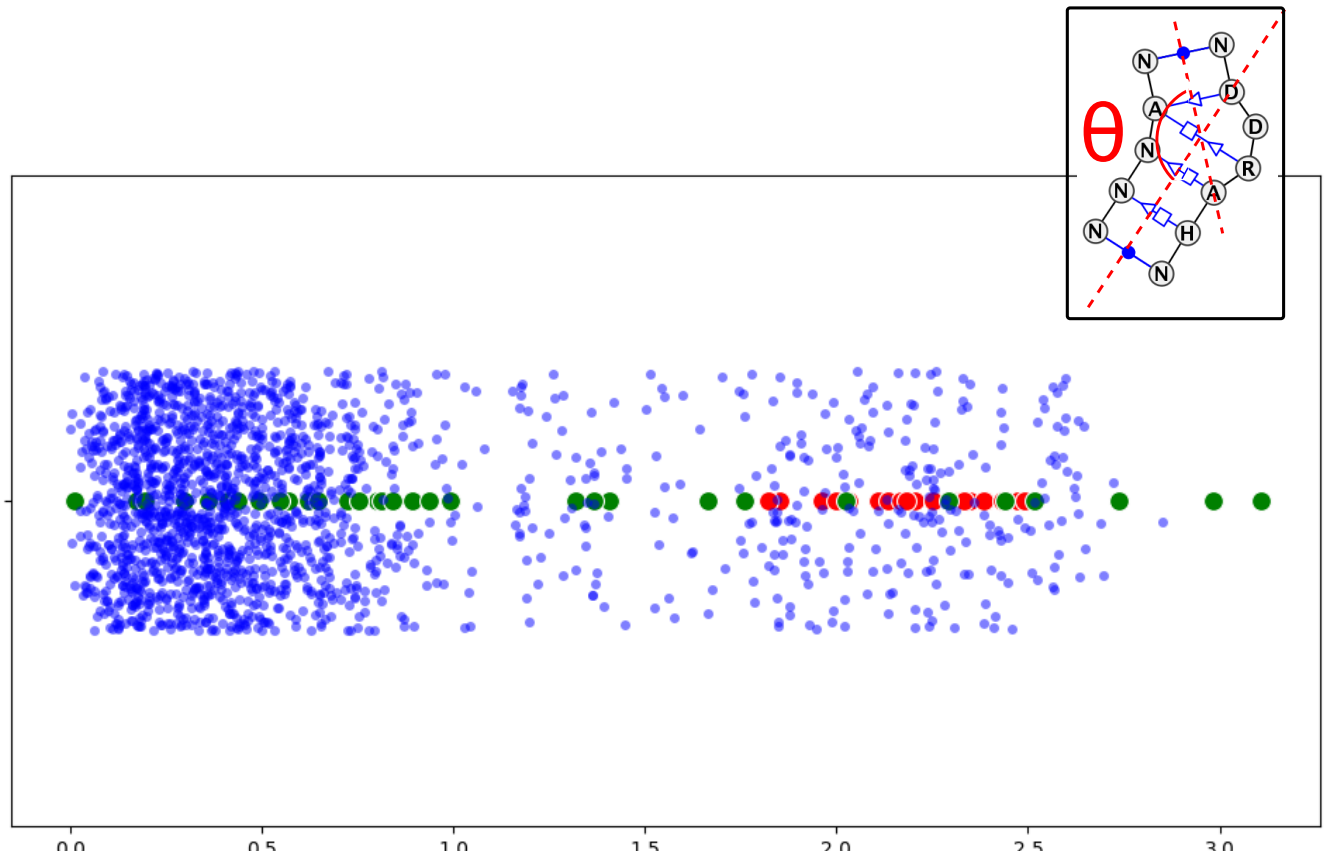
- Thresholds on neighborhoods: $T^L = 20$, $T^E = 4$ and $T^G = 20$.
- Used metrics: Sensitivity $= \frac{TP}{P}$ and Specificity $= \frac{TN}{N}$

Requested motif



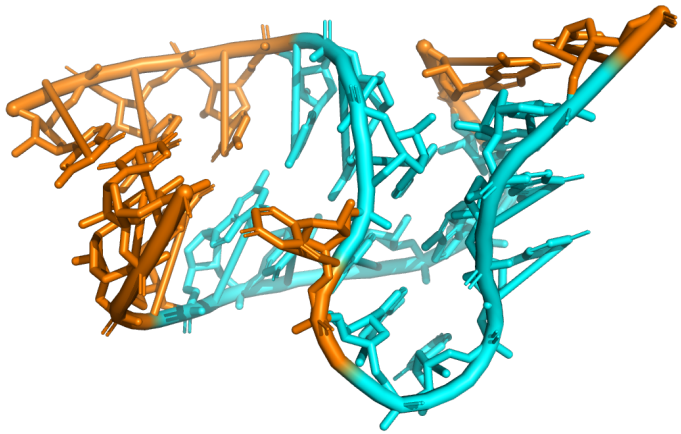


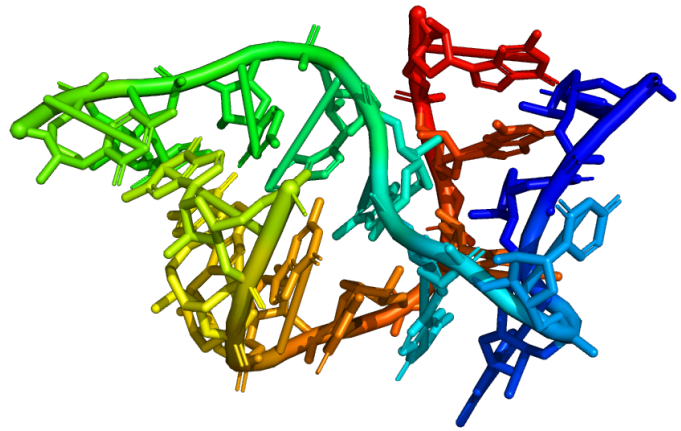
What about found motifs that are not labeled as Kink-Turns ?

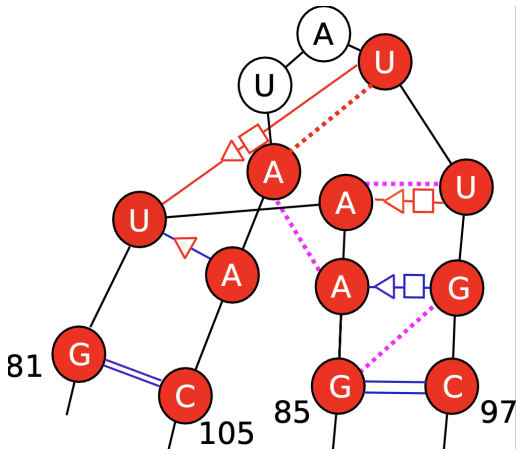


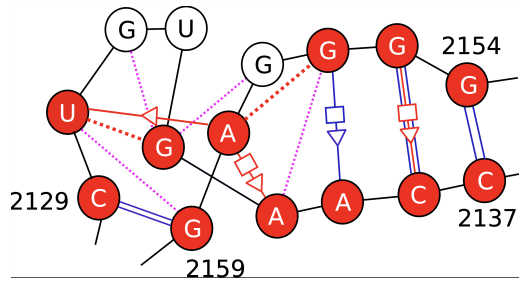
- Some of our motifs angles in green are in the range of the Kink-Turn angles in red.

Suggestions of new motifs using our methods



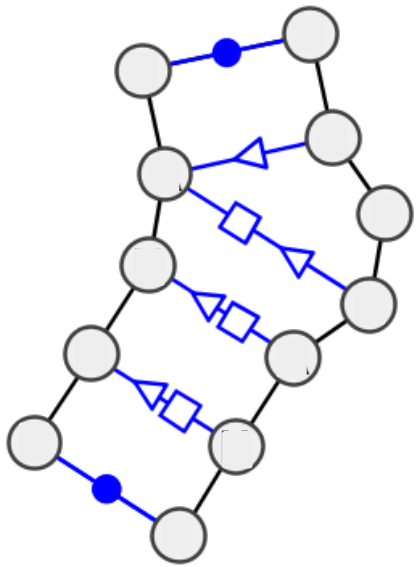




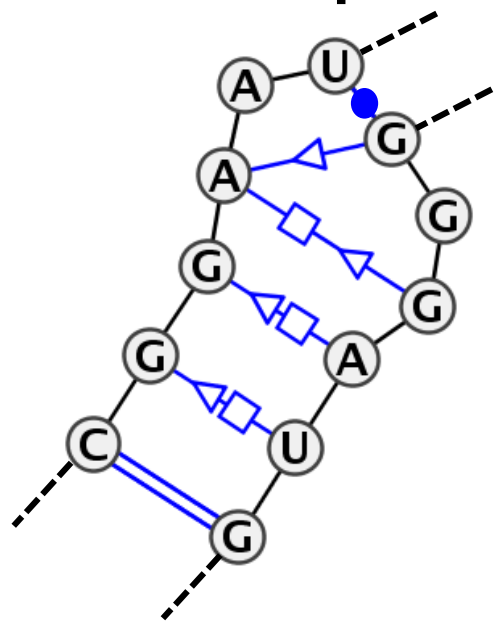


Case study: A Kink-Turn that we missed

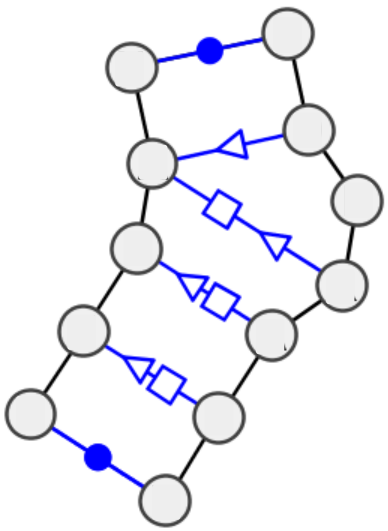
G_P

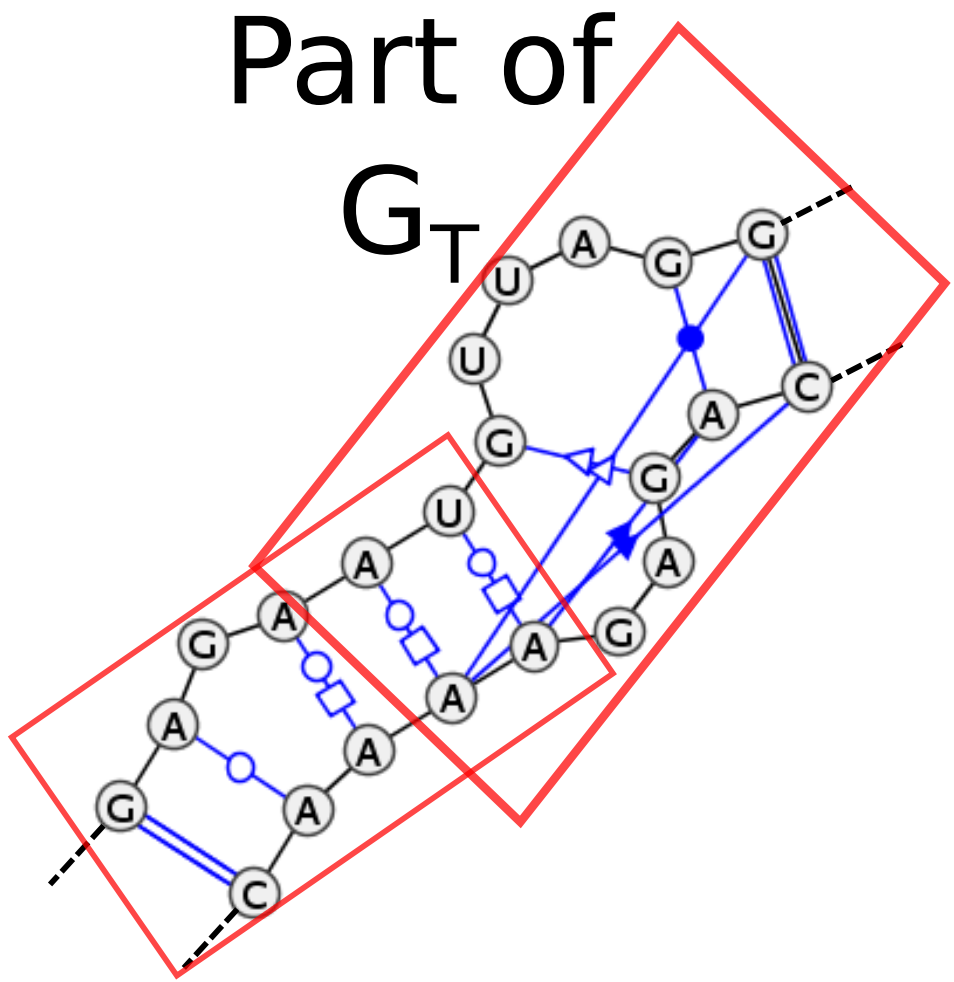


Part of G_T



Case study: Kink-Turn that we missed but that we can hope to cover with multiple mappings





Conclusion and future work

- We proposed an exact sampling solution for the Fuzzy Monotonous Subgraph Isomorphism Problem.
- Complexity is rooted in the treewidth of the pattern graph:

$$O\left(knt + kn^{\phi+1}\right)$$

- We used isostericity, missing bonds and gaps to catch a wide variety of RNA motifs as observed on the Kink-Turn.

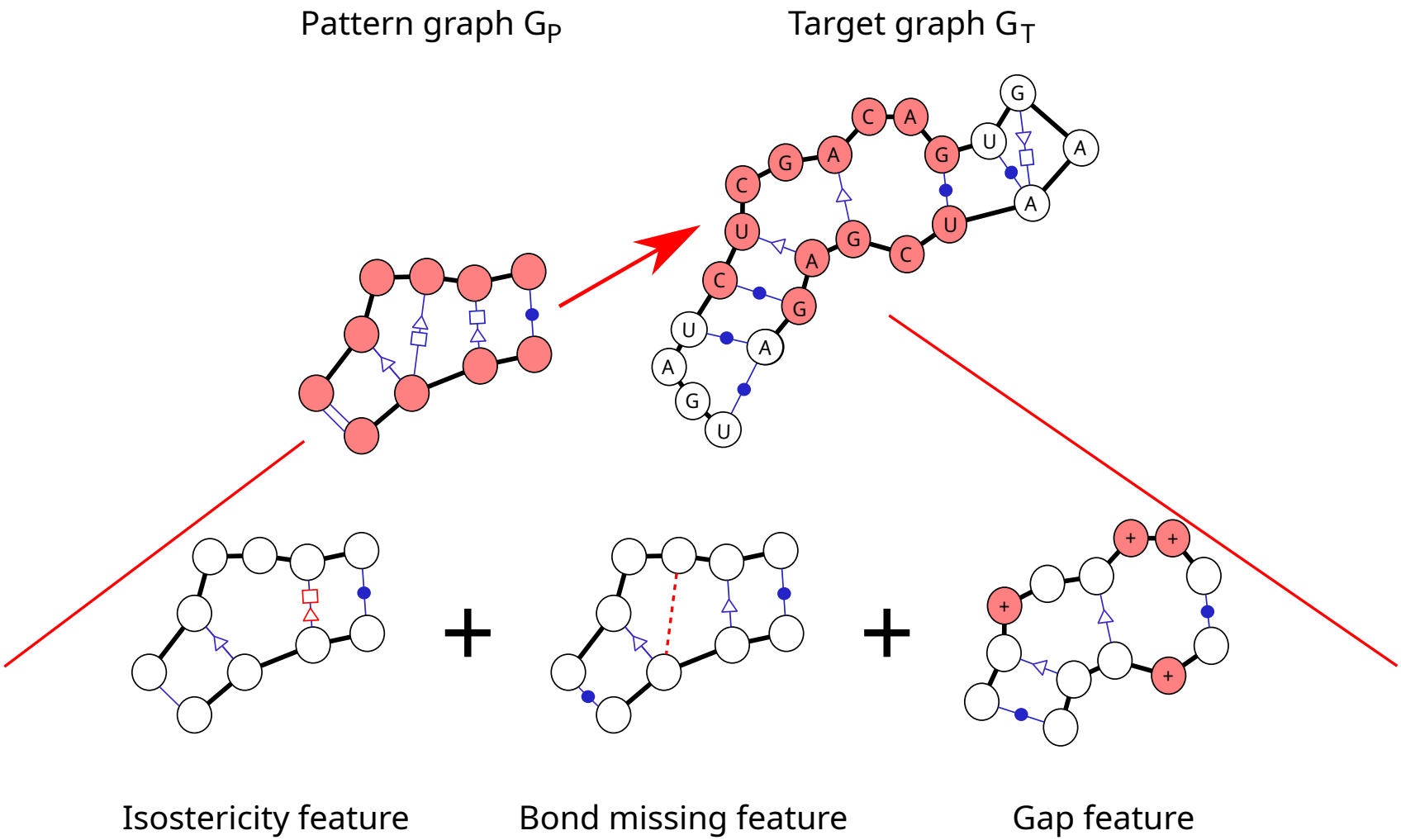
Future work:

- Further evaluate the efficiency of FuzzTree on diverse RNA modules
- Possibility to introduce new metrics without additional work
- Discover unknown RNA motifs unlisted until now thanks to our neighborhoods

Acknowledgements

You!

FuzzTree


Pattern graph G_P
Target graph G_T
Isostericity feature
Bond missing feature
Gap feature

Typical treewidth of Kink-Turn motifs

Number of Kink-Turn instances	Treewidth
50	2
21	3

- Complexity:

$$O\left(knt + kn^{\phi+1}\right)$$

- k, number of nodes in G_P.
- n, number of nodes in G_T.
- ϕ, treewidth of G_P.
- t, number of samples.

Isostericity computation

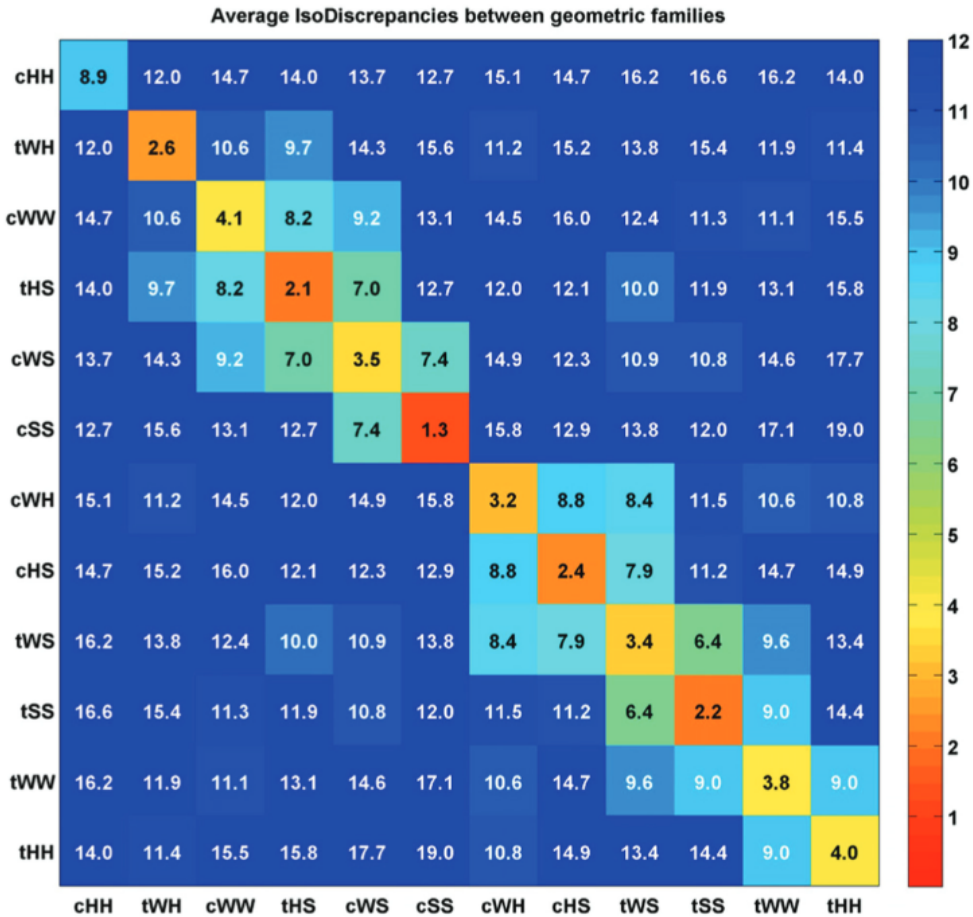


Figure adapted from Zirben al, 2009

Sampling usage and interests

Definition 3.1 (Multidimensional Boltzmann distribution): The probability P to sample a graph G of G_P in G_T depends on its (pseudo-)**energy** E:

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

With E a weighted combination over a collection of features $\{F_i\}$ of interest:

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

$w_1, w_2 \dots$ are real-valued weights.

- In our case, features $\{F_i\}$ quantify the distance between the pattern graph G_P and the mapping found in G_T.