

RNAPuzzler II

Daniel Gerighausen

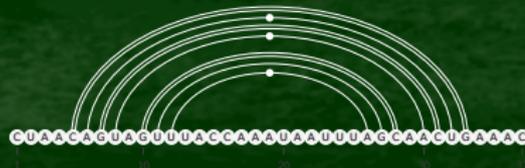
Image and Signal Processing Group
Leipzig University

Wintersemester 2016



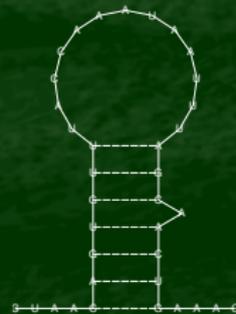
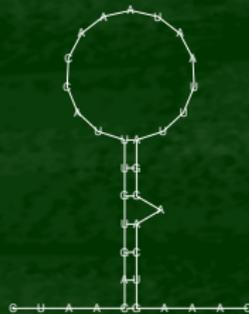
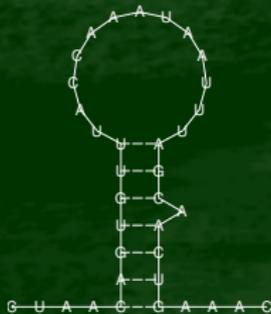
Introduction

- RNA secondary structure is a tree
 - loops are nodes, stems are edges
 - it is planar without pseudoknots
- it is possible to draw RNAs as a planar secondary structure!



Requirements

- Backbone and Basepair distances are independent
- the exterior loop is a straight line
- Bulges with only one free Base do not Bend the stem
- the visualization should be planar
- option to modify the drawing during post-processing



Basic Implementation

turtle graphic algorithm with affine coordinates

- the position of each base is calculated in relation to its predecessor with an angle and a specific distance
- structure detection for each base on the fly
- loop calculation with Newton-Raphson to calculate an appropriate radius
- special structures like mini-exterior loops or lonely basepairs need separate treatment

Template Based drawing

Drawing RNAs with drawing constraints?

- Similar folded RNAs can be visualized very differently
- Easier perception of differences between similar RNAs
- Adjusting the drawing to user's preferences
- First implementation to show post processing capabilities

Template Based drawing II

Preprocessing

- Using infernal's calign to create constraints for RNAfold
- Covariance model of the RNA family is necessary
- Filter constraints created by calign to make it compatible with RNAfold
- Remove insertions introduced by calign

Template Based drawing III

Config File

- RNAplot uses a config file for drawing constraints
- Each stem and each loop is identified by a variable
- Each line describes an angle between two stems in a loop
- Example: A B C $\frac{\pi}{2}$
- RNAplot identifies the loops and stems during the drawing process on the fly
- Validation check before applying the config

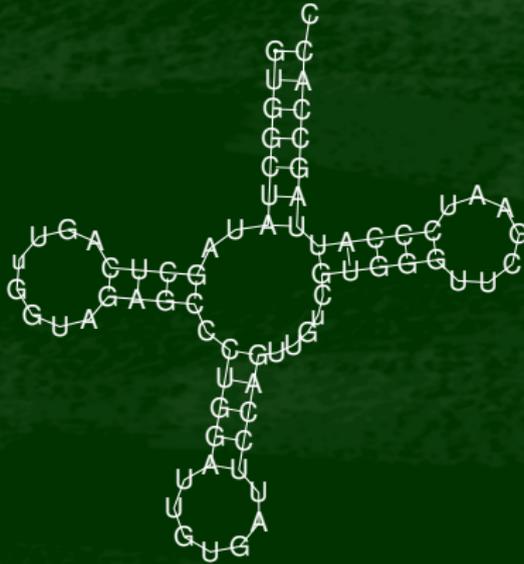
Template Based drawing IV

Applying Rules

- Due to the affine coordinates only the coordinates of bases in the loops have to be modified
- Calculate the radius for each fixed partial circle of the loop
 - Take the maximum of them
- Bend the backbone distances in all other partial circles to this radius
- Unknown stems are inserted into the constrained drawing

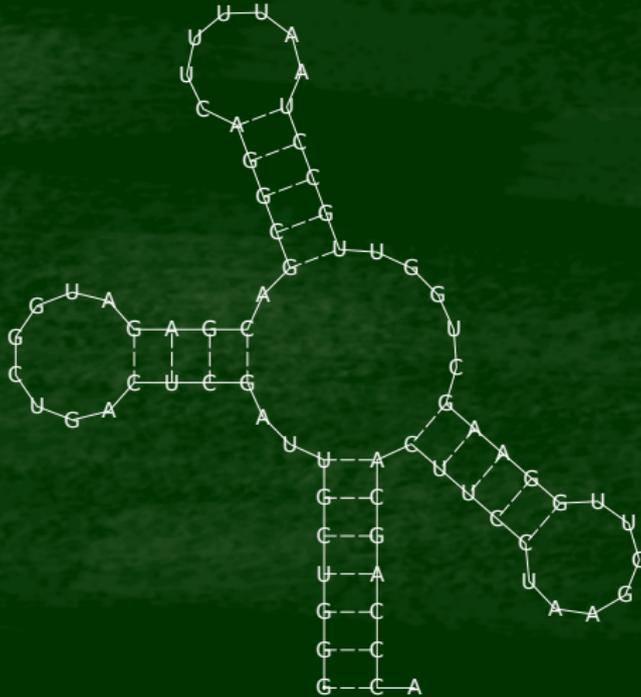
Results tRNA

Naview algorithm with folding constraints



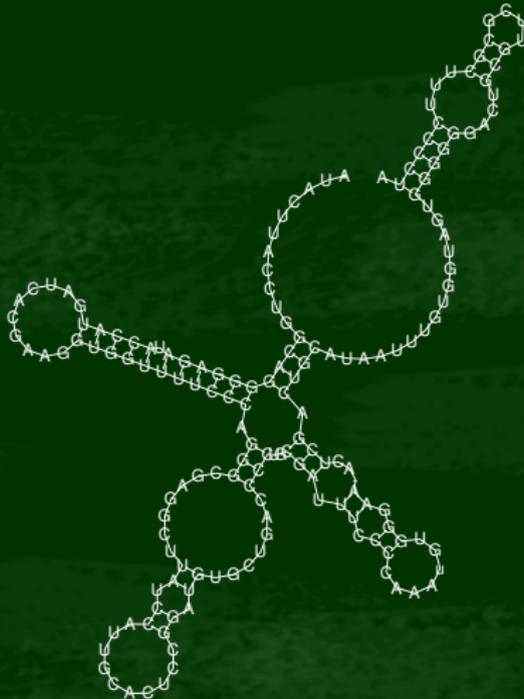
Results tRNA

RNApuzzler algorithm with folding constraints



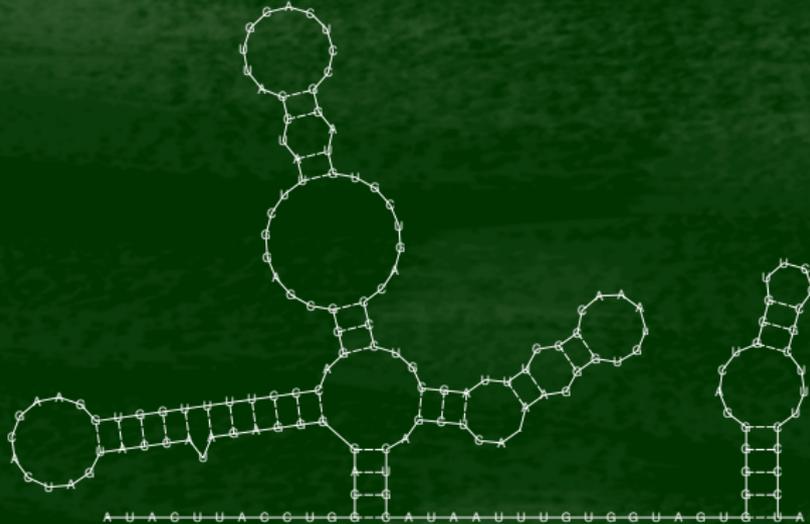
Results UI RNA

Naview algorithm with folding constraints



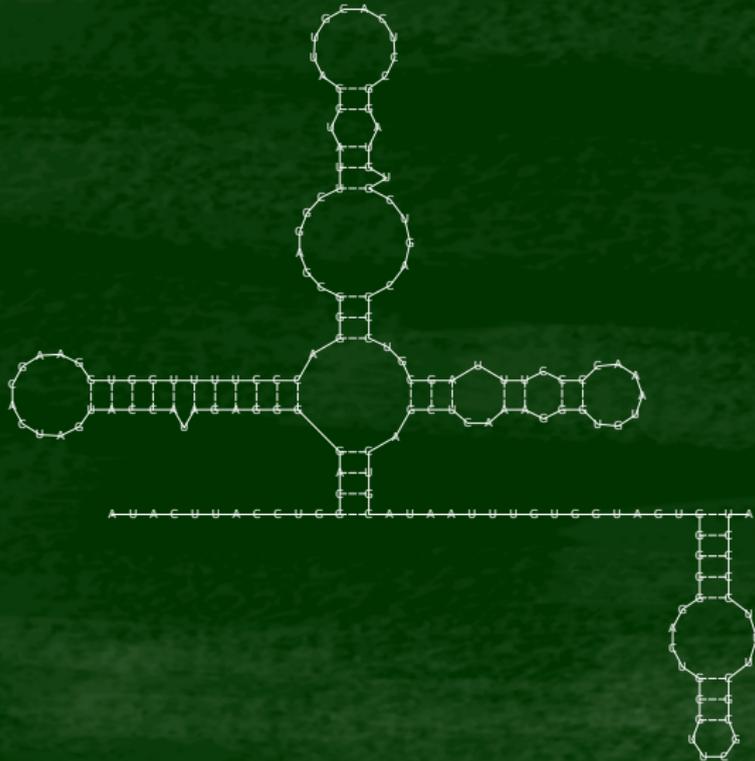
Results UI RNA

RNApuzzler algorithm with folding constraints



Results UI RNA

RNApuzzler algorithm with drawing constraints



Conclusion

RNApuzzler provides:

- an easy method to create constrained RNA drawings
- a visual assistance pipeline to compare RNA structures in their families
- user assisted planar RNA visualizations
- possibilities to modify the RNA drawing style at runtime

Future Work

- resolve the remaining intersection possibilities
- implement more general configurations to define RNA drawing styles without folding constraints
- user interface for designing RNA drawing configurations

Thanks to Peter and Dirk!
Thanks to Daniel Alexander!

Thank you for your attention!

Questions?

