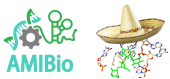
RNA *in silico* design with Infrared framework

Hua-Ting Yao

Theoretical Biochemistry Group (TBI), University of Vienna

Vienna, Austria

tbi



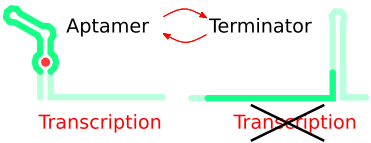
RNA (structural) design

- Finding RNA sequences with desired function

  → RNA vaccines, RNA drugs, therapies based on RNA . . .

- Structure ↔ Function

  → Off Riboswitch



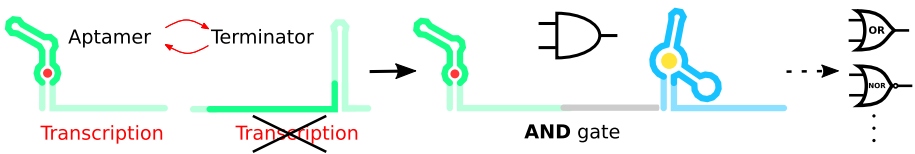
RNA (structural) design

- Finding RNA sequences with desired function

  → RNA vaccines, RNA drugs, therapies based on RNA . . .

- Structure ↔ Function

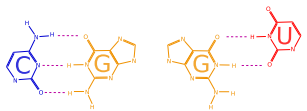
  → Off Riboswitch → Riboswitch **AND** (G. Domin *et al.*, 2017)

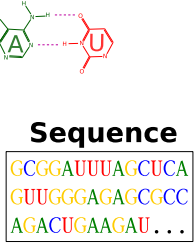


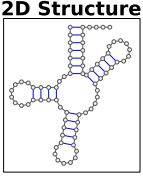
Outline

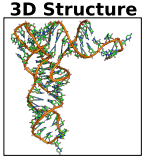
1. Brief introduction to RNA structural bioinformatics
2. Infrared in RNA design
3. Example 1: exonuclease-resistant RNA (xrRNA)
4. Example 2: small RNA - mRNA interaction

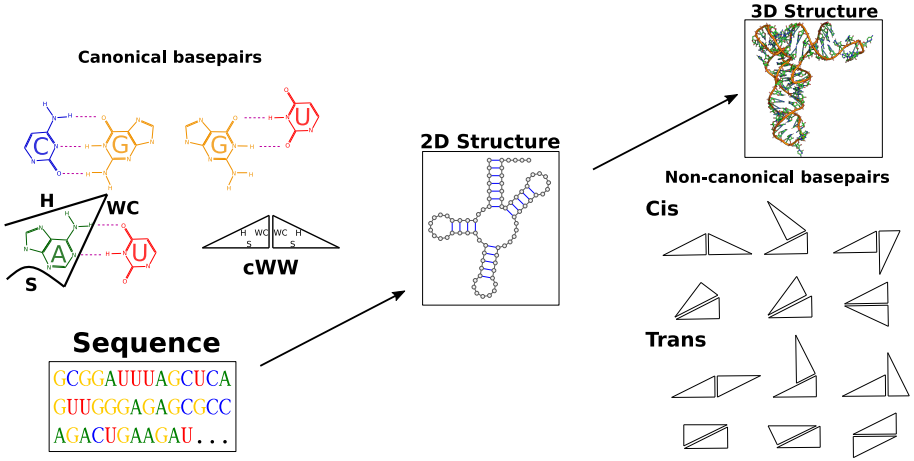
RNA sequence and structures



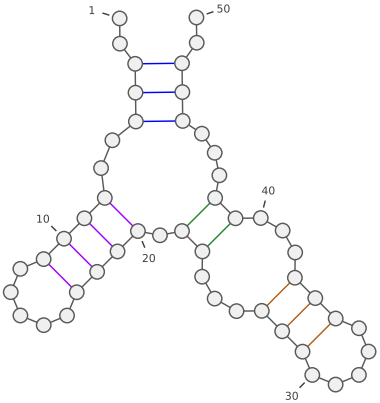


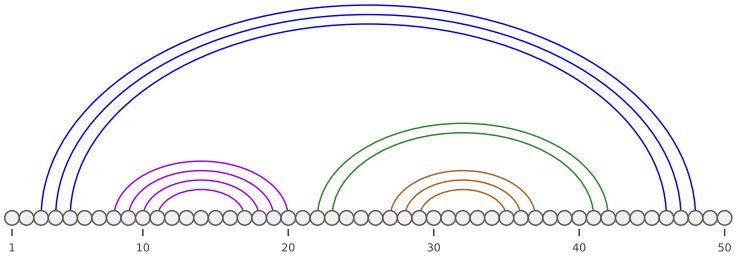





Canonical basepairs
H
WC
S
cWW
Sequence
GCGGAUUUAGCUCA
GUUGGGAGAGCGCC
AGACUGAAGAU...
2D Structure
3D Structure
Non-canonical basepairs
Cis
Trans

Representations of RNA structures

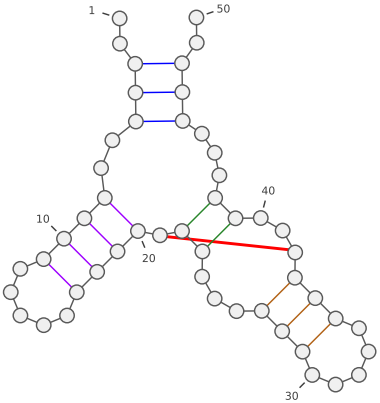


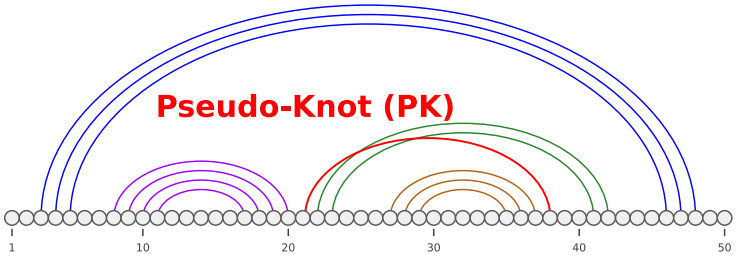


••(((••((((•••••))))•((•••(((•••••)))•••))•••)))••

Motzkin words $\rightarrow$ $\sim \frac{3^n}{n\sqrt{n}}$ secondary structures

Representations of RNA structures

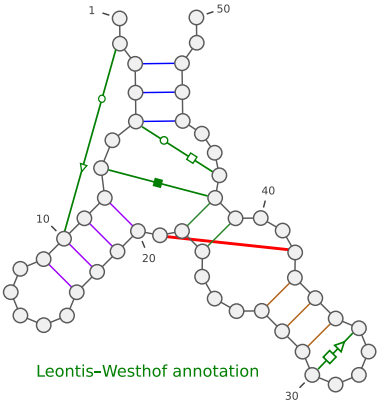




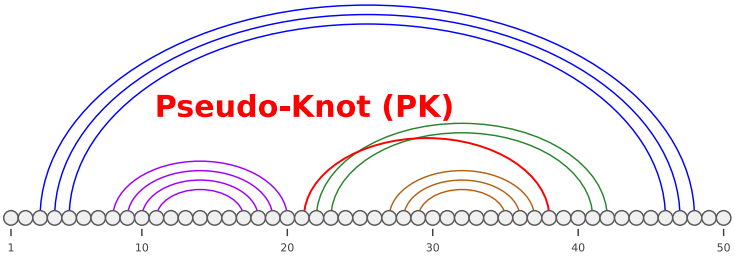
••(((••((((•••••))))[(((•••(((•••••)))]••)))•••)))••

Motzkin words $\rightarrow \quad \sim \frac{3^n}{n\sqrt{n}}$ secondary structures

Representations of RNA structures



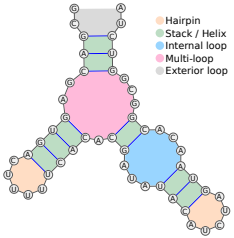
Leontis–Westhof annotation



••(((••((((•••••))))[(((•••(((•••••)))]••)))•••)))••

Motzkin words $\rightarrow$ $\sim \frac{3^n}{n\sqrt{n}}$ secondary structures

Thermodynamic model



- `ViennaRNA (1994 - )`

- Free energy $\mathcal{E}$

$$\mathcal{E}: \quad \Sigma^* \times \mathcal{S} \to \mathbb{R}, \quad (w, S) \mapsto \mathcal{E}(w, S)$$

w: sequence, S: structure

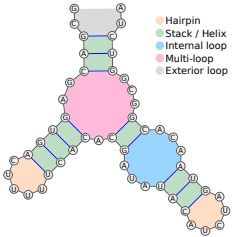
$$\mathcal{E}(w,S) = \Delta\Big(\quad\Big) + \Delta\Big(\quad\Big) + \Delta\Big(\quad\Big) + \cdots$$

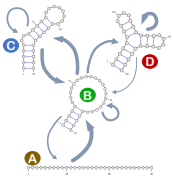
- Optimal conformation (Minimum Free-Energy, MFE)

  → secondary: $\mathcal{O}(n^3)$

  → pseudo-knotted: NP-hard

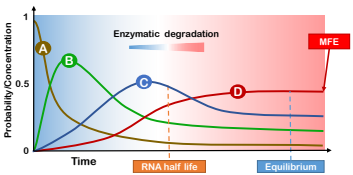
Thermodynamic model





A – Kinetic Landscape

Continuous-time Markov chain



B – Evolution of concentrations

(borrowed from Y. Ponty)

- `ViennaRNA (1994 - )`

- Free energy $\mathcal{E}$

$$\mathcal{E}: \quad \Sigma^* \times \mathcal{S} \to \mathbb{R}, \quad (w, S) \mapsto \mathcal{E}(w, S)$$

w: sequence, S: structure

$$\mathcal{E}(w,S) = \Delta\Big(\quad\Big) + \Delta\Big(\quad\Big) + \Delta\Big(\quad\Big) + \cdots$$

- Optimal conformation (Minimum Free-Energy, MFE)

  → secondary: $\mathcal{O}(n^3)$

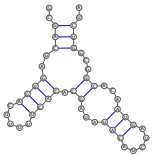
  → pseudo-knotted: NP-hard

- Boltzmann equilibrium, kinetic, ...

RNA Structural Design

- Positive Design: Compatibility with one or few target structure(s)
  → optimize affinity towards given target(s) *i.e.* free-energy

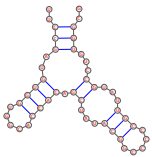
w_1



-4.3

- Negative Design: Avoidance of (exponential) unwanted structures
  → specificity towards given targets *i.e.* Minimum Free Energy (MFE)

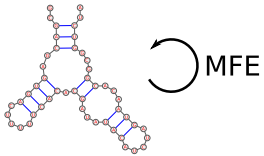
w_2



-6.7

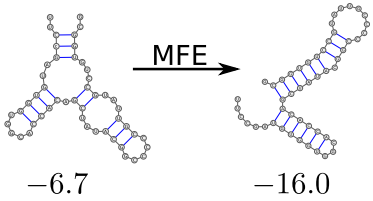
- Positive Design: Compatibility with one or few target structure(s)
  → optimize affinity towards given target(s) *i.e.* free-energy

w_1



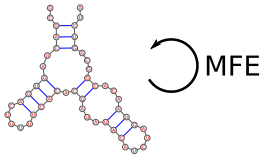
- Negative Design: Avoidance of (exponential) unwanted structures
  → specificity towards given targets *i.e.* Minimum Free Energy (MFE)

w_2



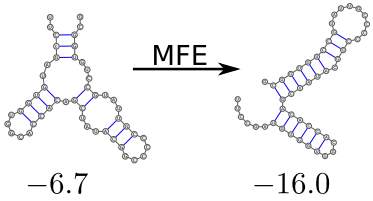
- Positive Design: Compatibility with one or few target structure(s)
  → optimize affinity towards given target(s) *i.e.* free-energy
  - Constraints: basepair compatibility, sequence pattern . . .
  - (Weight) Functions: free-energy, GC-content . . .
  - Sampling

w_1



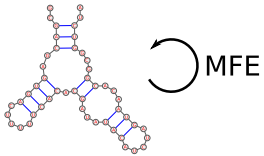
- Negative Design: Avoidance of (exponential) unwanted structures
  → specificity towards given targets *i.e.* Minimum Free Energy (MFE)

w_2



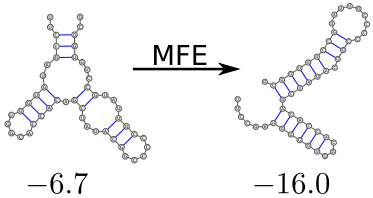
- Positive Design: Compatibility with one or few target structure(s)
  → optimize affinity towards given target(s) *i.e.* free-energy
  - Constraints: basepair compatibility, sequence pattern . . .
  - (Weight) Functions: free-energy, GC-content . . .
  - Sampling

w_1



- Negative Design: Avoidance of (exponential) unwanted structures
  → specificity towards given targets *i.e.* Minimum Free Energy (MFE)
  - Objective function: MFE, structure ensemble . . .
  - Optimization

w_2

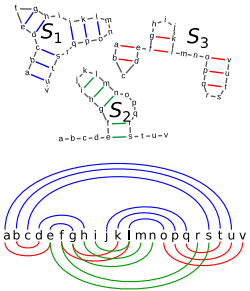


Infrared

General and efficient framework for weighted constraint satisfaction problem (CSP)

https://amibio.gitlabpages.inria.fr/Infrared/

Input

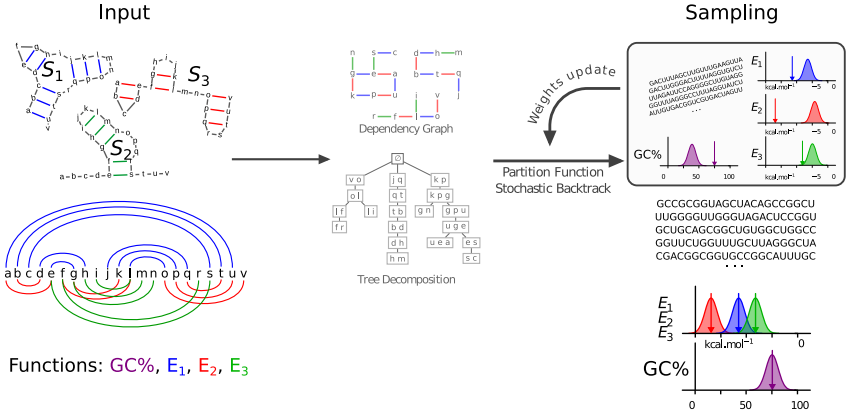


Functions: GC%, E_1, E_2, E_3

(modified from S. Hammer *et al*, 2019)

General and efficient framework for weighted constraint satisfaction problem (CSP)

https://amibio.gitlabpages.inria.fr/Infrared/

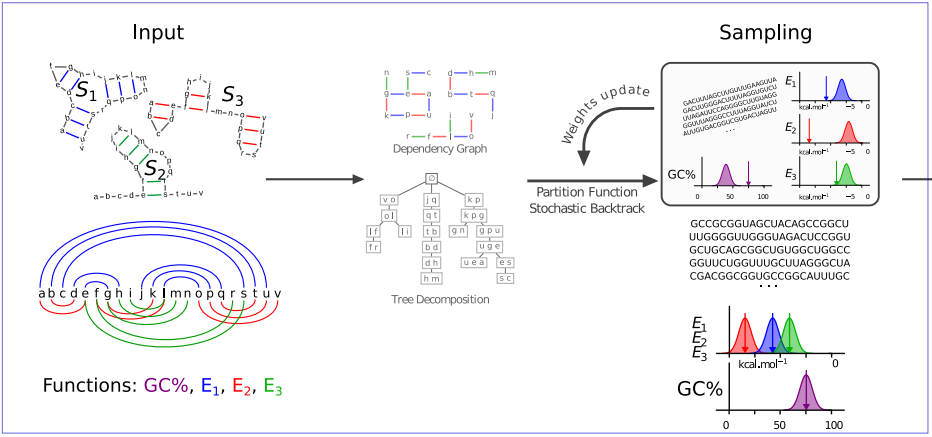


(modified from S. Hammer *et al*, 2019)

Infrared

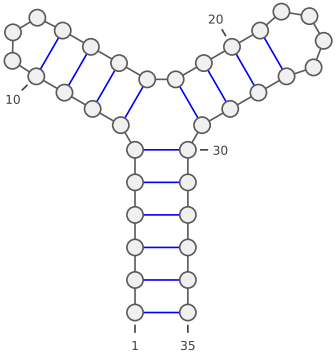
General and efficient framework for weighted constraint satisfaction problem (CSP)

https://amibio.gitlabpages.inria.fr/Infrared/



(modified from S. Hammer *et al*, 2019)

Single structure design

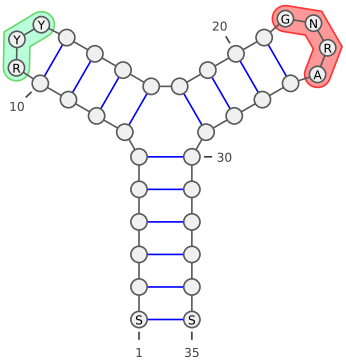



```
import infrared as ir
import infrared.rna as rna

model = ir.Model(35, 4)

target = "((((((((((...))))))((((....))))))))))"
model.add_constraints(rna.BPComp(i, j)
    for (i, j) in rna.parse(target))

sampler = ir.Sampler(model)
samples = [sampler.sample() for _ in range(1000)]
```

0 : A 1 : C 2 : G 3 : U

AU, CG, · · ·

Single structure design



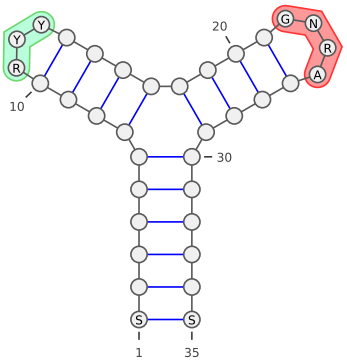

```
import infrared as ir
import infrared.rna as rna

model = ir.Model(35, 4)

target = "((((((((((...))))))((((....))))))))))"
model.add_constraints(rna.BPComp(i, j)
    for (i, j) in rna.parse(target))

iupac_seq = "SNNNNNNNNNRYYNNNNNNNNGNRANNNNNNNNNS"
for i, x in enumerate(iupac_seq):
    model.add_constraints(
        ir.ValueIn(i, rna.iupacvalues(x)))


sampler = ir.Sampler(model)
samples = [sampler.sample() for _ in range(1000)]
```

0 : A 1 : C 2 : G 3 : U

AU, CG, · · ·

N : ACGU S : CG R : AG Y : CU

Single structure design



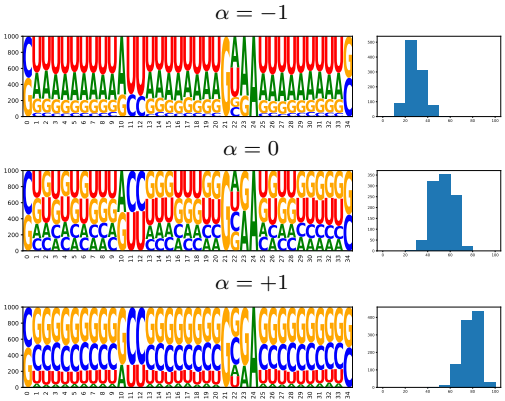

```
import infrared as ir
import infrared.rna as rna

model = ir.Model(35, 4)

target = "((((((((((...))))))((((....))))))))))"
model.add_constraints(rna.BPComp(i, j)
    for (i, j) in rna.parse(target))

iupac_seq = "SNNNNNNNNNRYYNNNNNNNNGNRANNNNNNNNNS"
for i, x in enumerate(iupac_seq):
    model.add_constraints(
        ir.ValueIn(i, rna.iupacvalues(x)))


sampler = ir.Sampler(model)
samples = [sampler.sample() for _ in range(1000)]
```

0 : A 1 : C 2 : G 3 : U

AU, CG, · · ·

N : ACGU S : CG R : AG Y : CU

Control GC



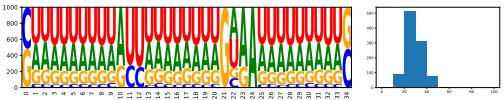
Method 1:

CG : 1 AU : 0

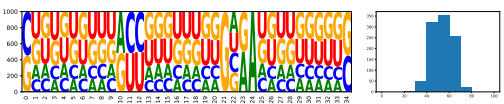
```
model.add_functions([rna.GCCont(i))
    for i in range(n)], 'gc')
model.set_feature_weight(α, 'gc')

sampler = ir.Sampler(model)
samples = [sampler.sample() for _ in range(1000)]
```

Control GC

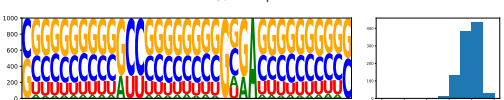
$\alpha = -1$



$\alpha = 0$



$\alpha = +1$



Method 1:

```
model.add_functions([rna.GCCont(i)) for i in range(n)], 'gc')
model.set_feature_weight(α, 'gc')

sampler = ir.Sampler(model)
samples = [sampler.sample() for _ in range(1000)]
```

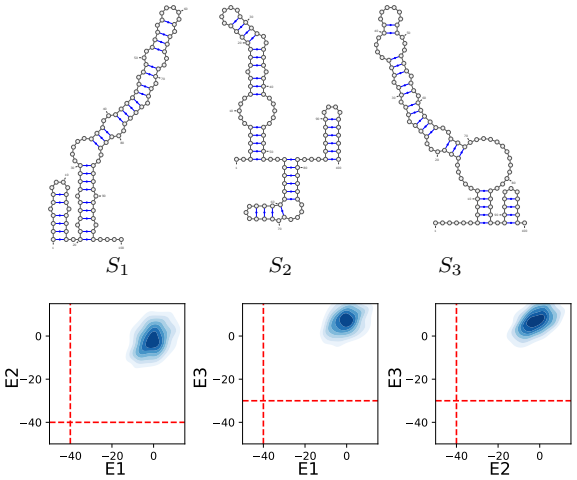
CG : 1 AU : 0

Method 2 (Targeted sampling):

```
sampler = ir.Sampler(model)
sampler.set_target( 0.75 * n, 0.01 * n, 'gc' )
samples = [sampler.targeted_sample()
    for _ in range(1000)]
```

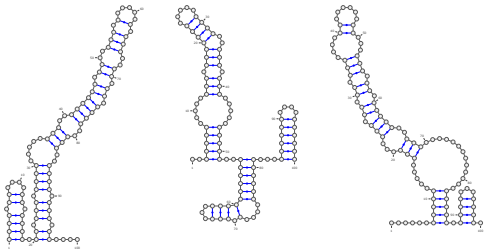
Automatically update α

Multidimensional Boltzmann sampling




```
model = ir.Model(n,4)

for k, target in enumerate(targets):
    bps = rna.parse(target)
    model.add_constraints(rna.BPComp(i, j)
        for (i, j) in bps)
```

Multidimensional Boltzmann sampling

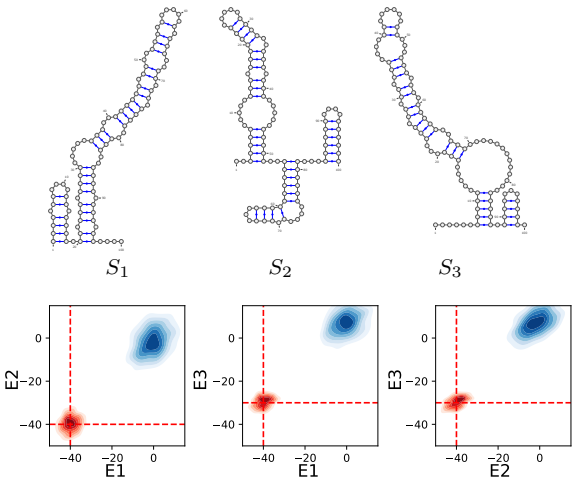



```
model = ir.Model(n,4)

for k, target in enumerate(targets):
    bps = rna.parse(target)
    model.add_constraints(rna.BPComp(i, j)
        for (i, j) in bps)

    model.add_functions([rna.BPEnergy(i, j)
        for (i, j) in bps], f'energy{k}')
```

Simplified energy model

Sample with simplified functions

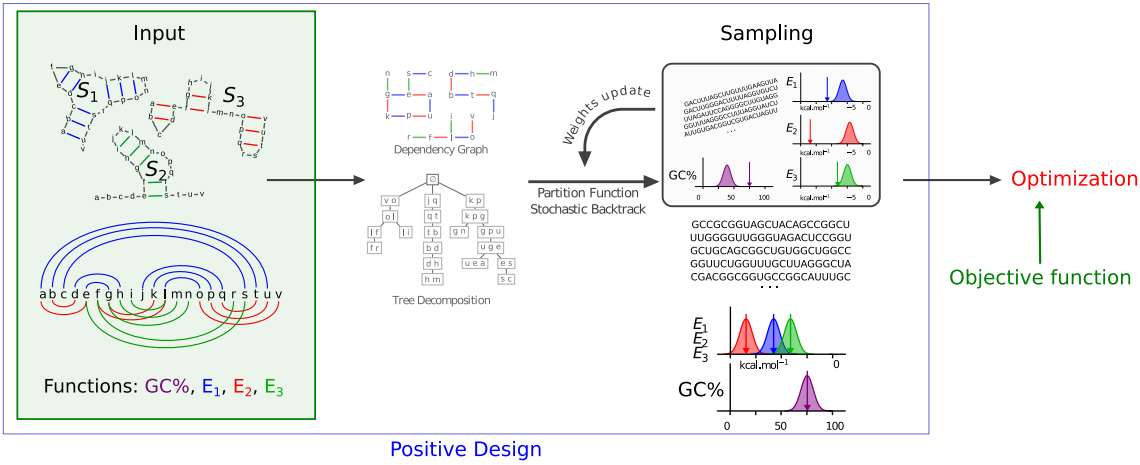
Multidimensional Boltzmann sampling




```
model = ir.Model(n,4)

for k, target in enumerate(targets):
    bps = rna.parse(target)
    model.add_constraints(rna.BPComp(i, j)
        for (i, j) in bps)

                                  Simplified energy model

    model.add_functions([rna.BPEnergy(i, j)
        for (i, j) in bps], f'energy{k}')


for k, target in enumerate(targets):
    model.add_feature(f'Energy{k}', f'energy{k}',
        lambda sample, target=target:
            energy_of_struct(sample, target))

                    ViennaRNA energy model

sampler = ir.Sampler(model)
sampler.set_target( -40, 0.5, 'Energy0')
sampler.set_target( -40, 0.5, 'Energy1')
sampler.set_target( -30, 0.5, 'Energy2')
```

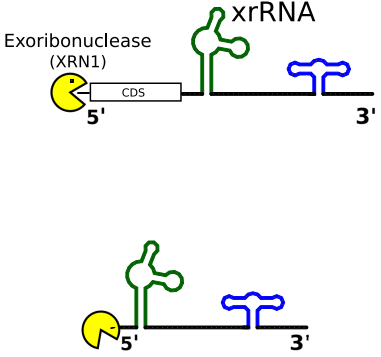
Sample with simplified functions, then target with complete and complex feature

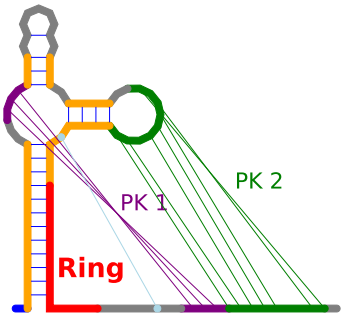
Summary of Infrared

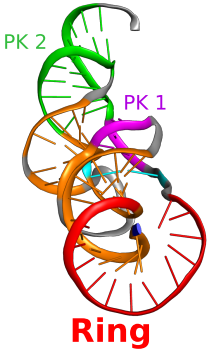


Positive Design

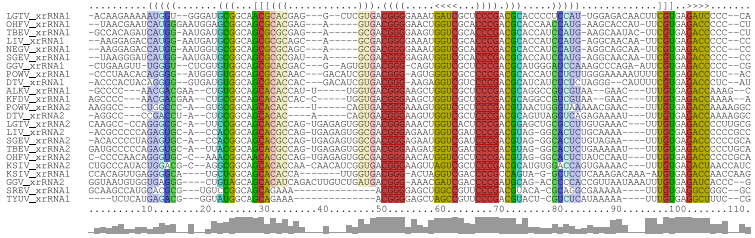
Example 1: exonuclease-resistant RNA (xrRNA)

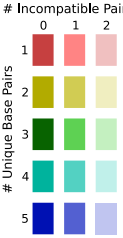


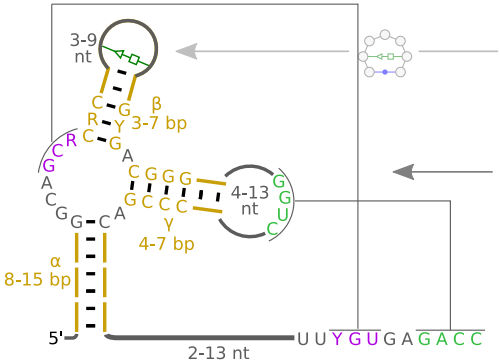




Example 1: Design of xrRNA

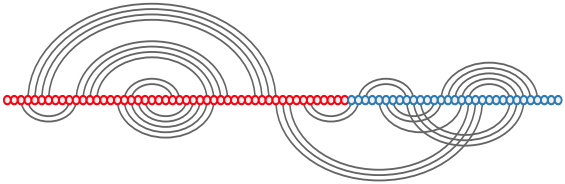


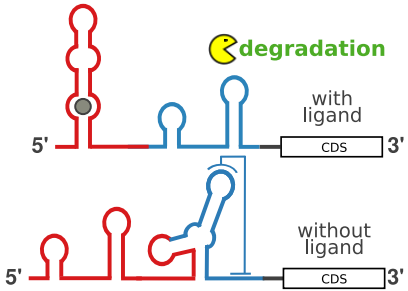




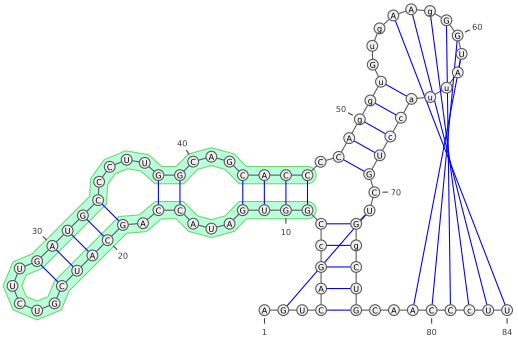
1. Sampling:
   - sequence conservation
   - varied length
2. Optimization:
   - 2D: ensemble defect
   - (Future work) PK: CParty (L. Trinity *et al*, 2024)

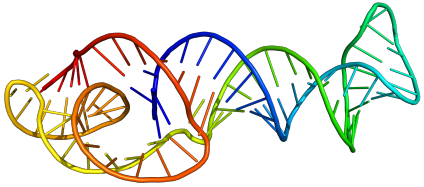
Example 1: xrRNA-riboswitch





Example 1: Aptamer insertion

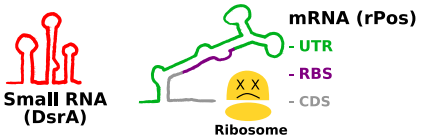




(Simulation with simRNA)

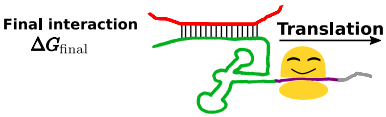
Experimental validation is ongoing

Example 2: small RNA - mRNA interaction

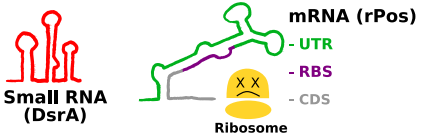


messenger RNA (mRNA):

1. **UTR**: Untranslated region
2. **RBS**: Ribosome binding site
3. **CDS: Coding sequence**



Example 2: small RNA - mRNA interaction

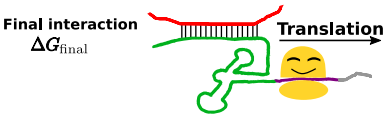


messenger RNA (mRNA):

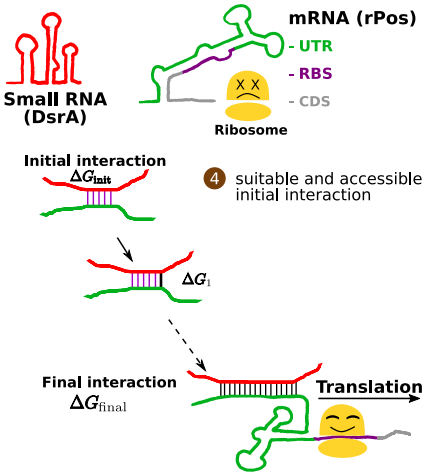
1. **UTR**: Untranslated region
2. **RBS**: Ribosome binding site
3. **CDS: Coding sequence**

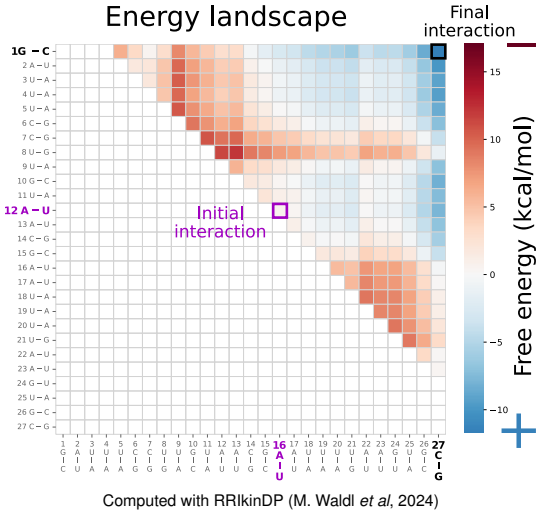
Design goals:

1. sRNA and mRNA should bind strongly
2. poor **RBS** accessibility w/o sRNA
3. in the bound state, **RBS** is highly accessible



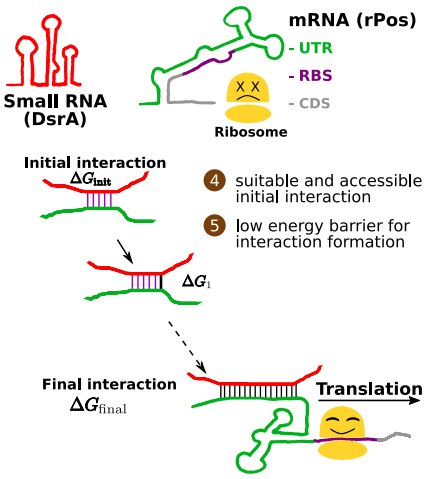
Example 2: small RNA - mRNA interaction

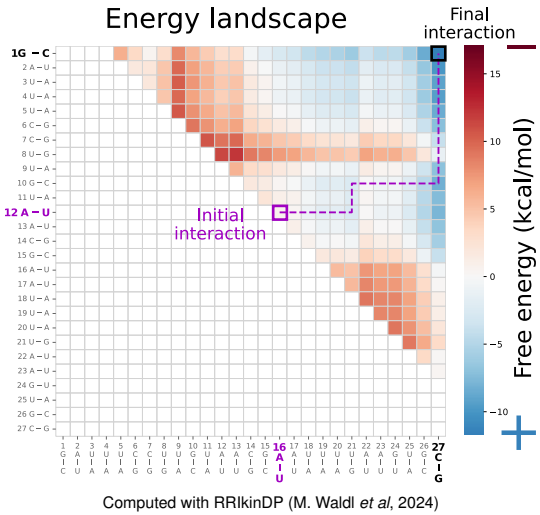




Computed with RRIkinDP (M. Waldl *et al*, 2024)

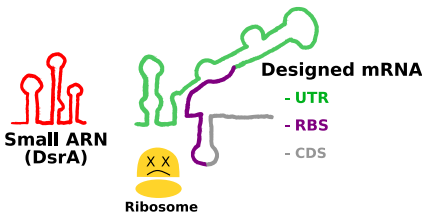
Example 2: small RNA - mRNA interaction





Computed with RRIkinDP (M. Waldl *et al*, 2024)

Example 2: Design of interaction (equilibrium only)

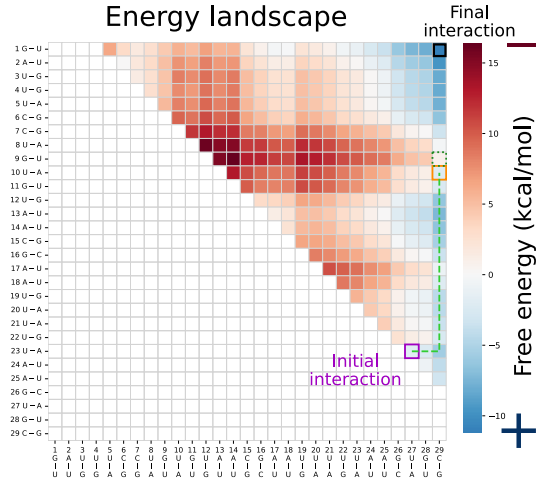


Design w/o kinetic:

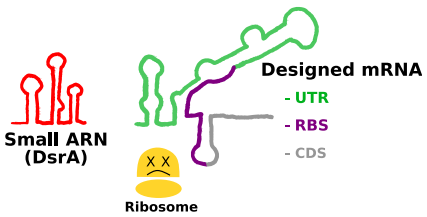
1. sRNA and mRNA should bind strongly $\approx -12.97\,\mathrm{kcal/mol}$
2. poor RBS accessibility w/o sRNA $\approx 12.29\,\mathrm{kcal/mol}$

V

3. in the bound state, RBS is highly accessible $\approx 5.01\,\mathrm{kcal/mol}$

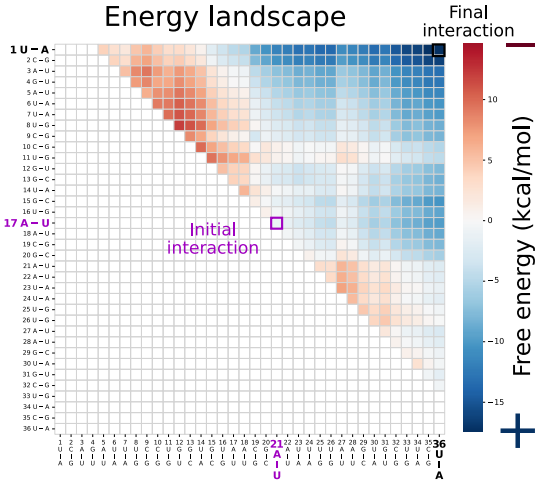


Example 2: Design of interaction with kinetic

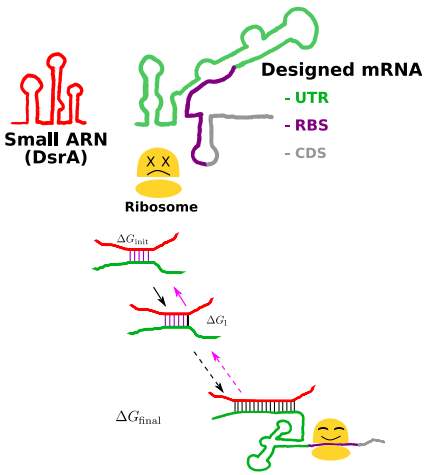


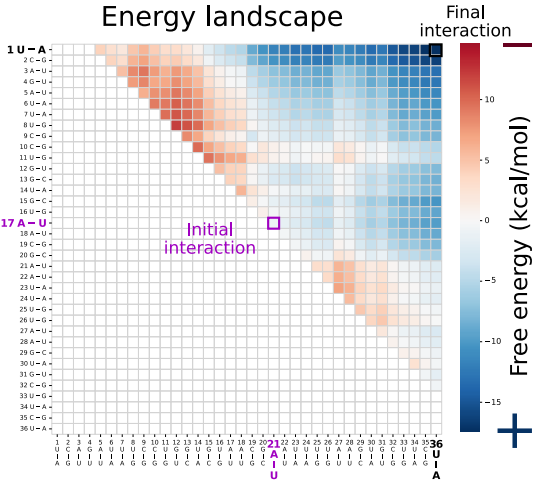
Design with kinetic:

1. sRNA and mRNA should bind strongly $\approx -17.00\,\mathrm{kcal/mol}$
2. poor RBS accessibility w/o sRNA $\approx 12.58\,\mathrm{kcal/mol}$
3. in the bound state, RBS is highly accessible $\approx 4.88\,\mathrm{kcal/mol}$
4. suitable and accessible initial interaction
5. low energy barrier for interaction formation



Example 2: Design of interaction with kinetic





Conclusion & Perspective

- Infrared: flexible and efficient framework for RNA design
  → Sequence (structure) alignment, phylogenetic reconstruction
  → Explore of 3D motif variations (T. Boury *et al*, 2023)

- Negative design via sampling (RNAPOND, HT Yao *et al*, 2021)

- Design beyond secondary structure: PK, 3D, interaction

Conclusion & Perspective

- Infrared: flexible and efficient framework for RNA design
  → Sequence (structure) alignment, phylogenetic reconstruction
  → Explore of 3D motif variations (T. Boury *et al*, 2023)

- Negative design via sampling (RNAPONDO, HT Yao *et al*, 2021)

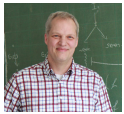
- Design beyond secondary structure: PK, 3D, interaction

- Design with different conditions thanks to latest ViennaRNA supports
  → some modifications (Y. Varenyk *et al*, 2023), mono-valent salt concentration (HT Yao *et al*, 2023)

- Near-structure design

Acknowledgment

R. Lorenz 

I. Hofacker 

P. Stadler 

Y. Ponty 

S. Will 

S. Berkemer 

M. Wolfinger 

L. Sidl 

tbi

J. Waldispühl 

B. Marchand 

M. Waldl 

V. Reinharz