Constraints in RNA Secondary structure prediction

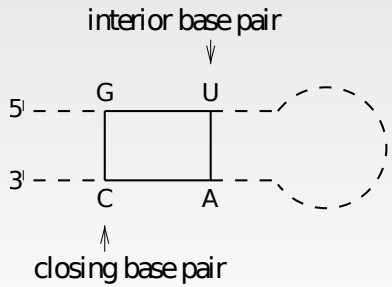
Ronny Lorenz
ronny@bioinf.uni-leipzig.de

Bioinformatik
University of Leipzig

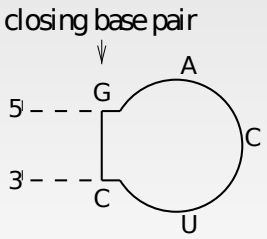
Leipzig, Germany, April 14, 2014

RNA Secondary structure prediction

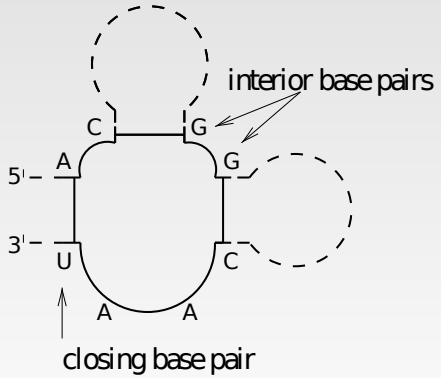
Secondary structures can be uniquely decomposed into loops



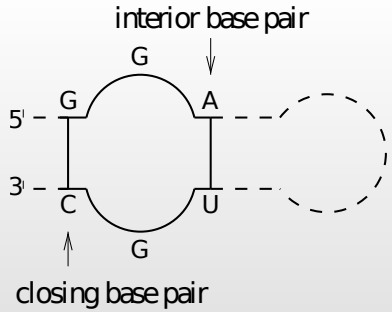
stacking pair



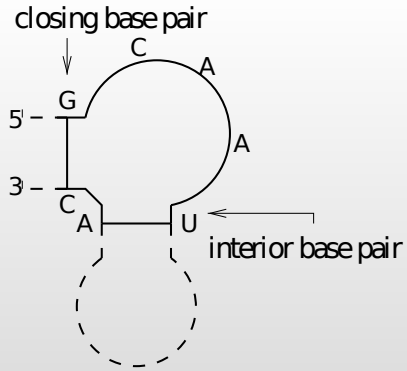
hairpin loop



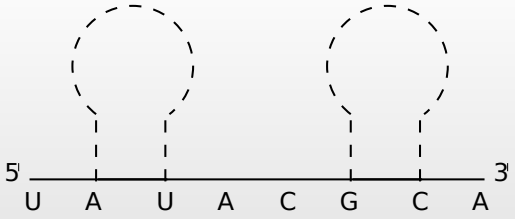
multi loop



interior loop

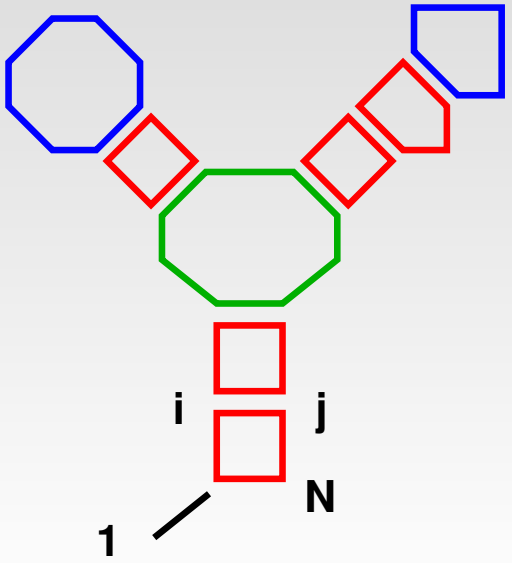


bulge



exterior loop

RNA Secondary structure prediction

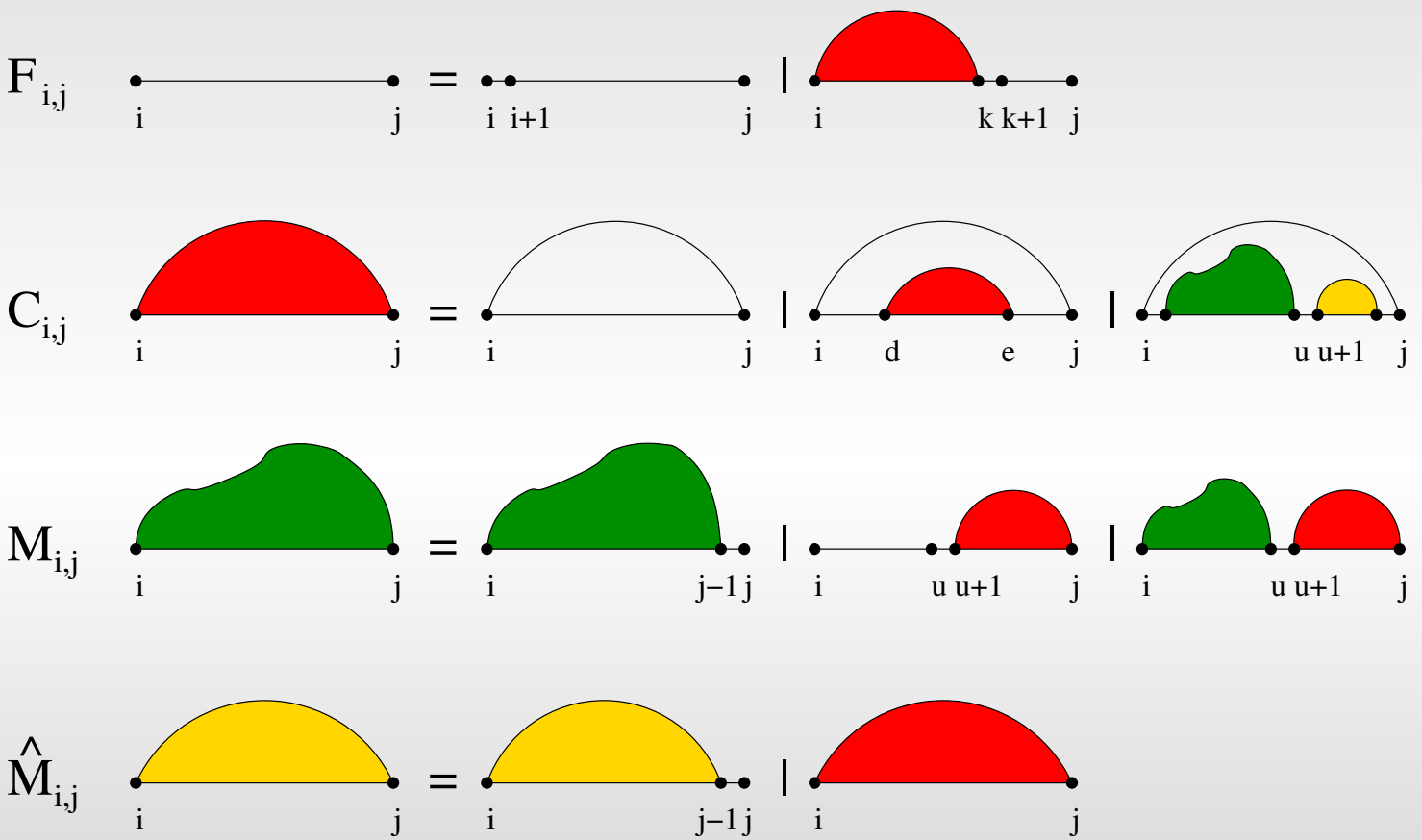


$$E(\mathcal{S}) = \sum_{L \in \mathcal{S}} E(L)$$

- The free energy of a secondary structure is the sum of the free energy of the loops its composed of
- Loop energies depend on loop type, loop size and sequence
- Energy parameters are measured experimentally or extrapolated by mathematical models

RNA Secondary structure prediction

Decomposition scheme



What is constraint folding

What happens during secondary structure prediction:

- decomposition scheme is applied to a sequence
- underlying energy model assign contributions to each decomposition
- algorithm finds e.g. an optimal structure (MFE) or adds up Boltzmann factors (PF)

What is constraint folding

What happens during secondary structure prediction:

- decomposition scheme is applied to a sequence
- underlying energy model assign contributions to each decomposition
- algorithm finds e.g. an optimal structure (MFE) or adds up Boltzmann factors (PF)

But:

- the energy model is not perfect
- experiment (e.g. SHAPE) may suggest differently to a prediction
- bound molecules (proteins, small ligands, etc.) prohibit certain structure elements and/or induce change in free energy

What is constraint folding

What happens during secondary structure prediction:

- decomposition scheme is applied to a sequence
- underlying energy model assign contributions to each decomposition
- algorithm finds e.g. an optimal structure (MFE) or adds up Boltzmann factors (PF)

But:

- the energy model is not perfect
- experiment (e.g. SHAPE) may suggest differently to a prediction
- bound molecules (proteins, small ligands, etc.) prohibit certain structure elements and/or induce change in free energy

Secondary structure constraints:

- disallow certain parses of the decomposition scheme (Hard Constraints)
  Example: exclude a (set of) nucleotide(s) from base pairing
- alter the energy contributions of the model (Soft Constraints)
  Example: add a bonus/malus when a nucleotide is considered unpaired

What is constraint folding

Hard Constraints allows for cutting out/ inserting[1] points in the secondary structure energy landscape

[1] circumvention of build-in constraints, e.g canonical base pairs

What is constraint folding

Hard Constraints allows for cutting out/ inserting[1] points in the secondary structure energy landscape

2

[1] circumvention of build-in constraints, e.g canonical base pairs

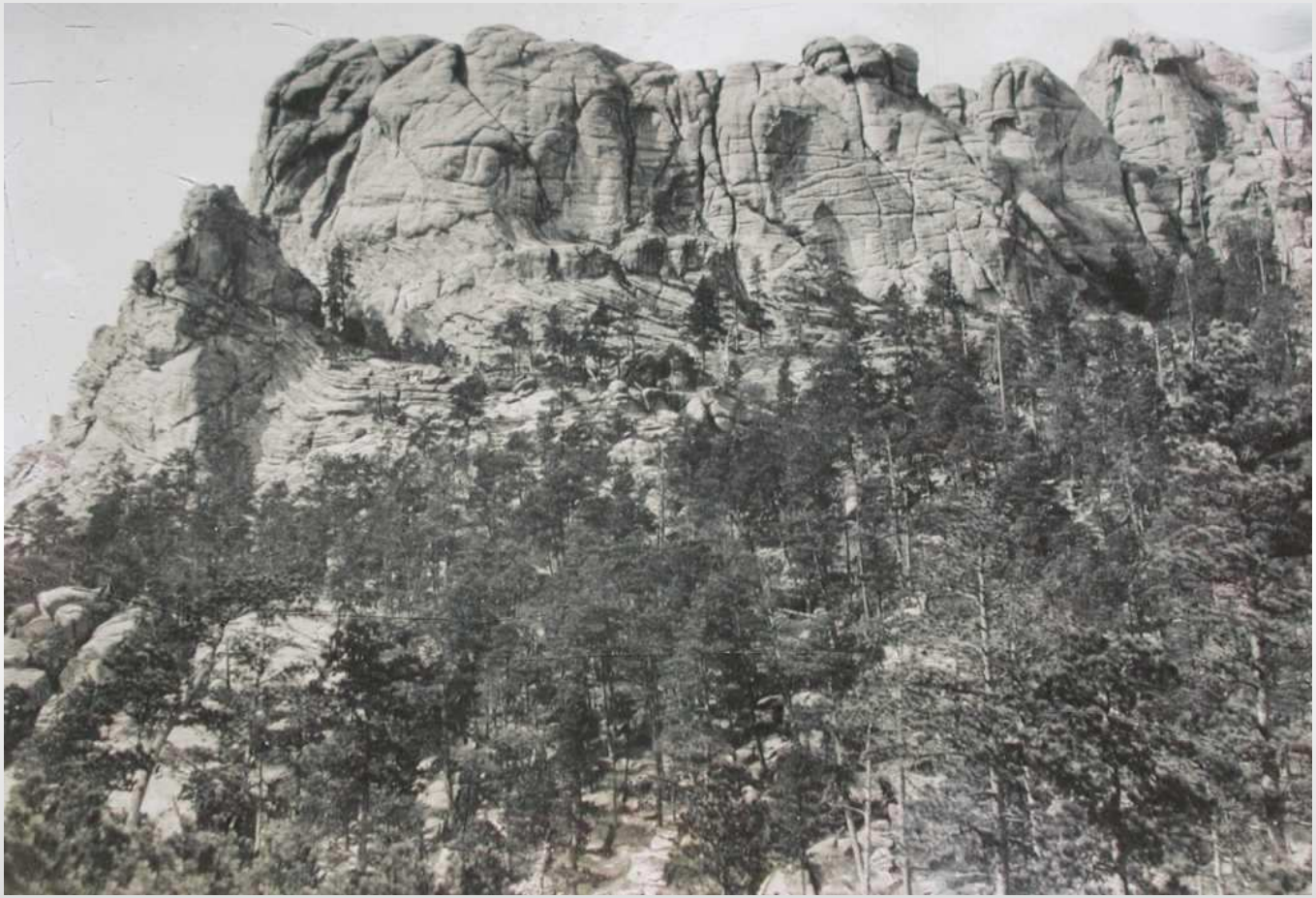
[2] Gobierno de Álvaro Colom, Guatemala

What is constraint folding

Soft Constraints allow for shifting points in the landscape up or down

What is constraint folding

Soft Constraints allow for shifting points in the landscape up or down



Mount Rushmore 1925

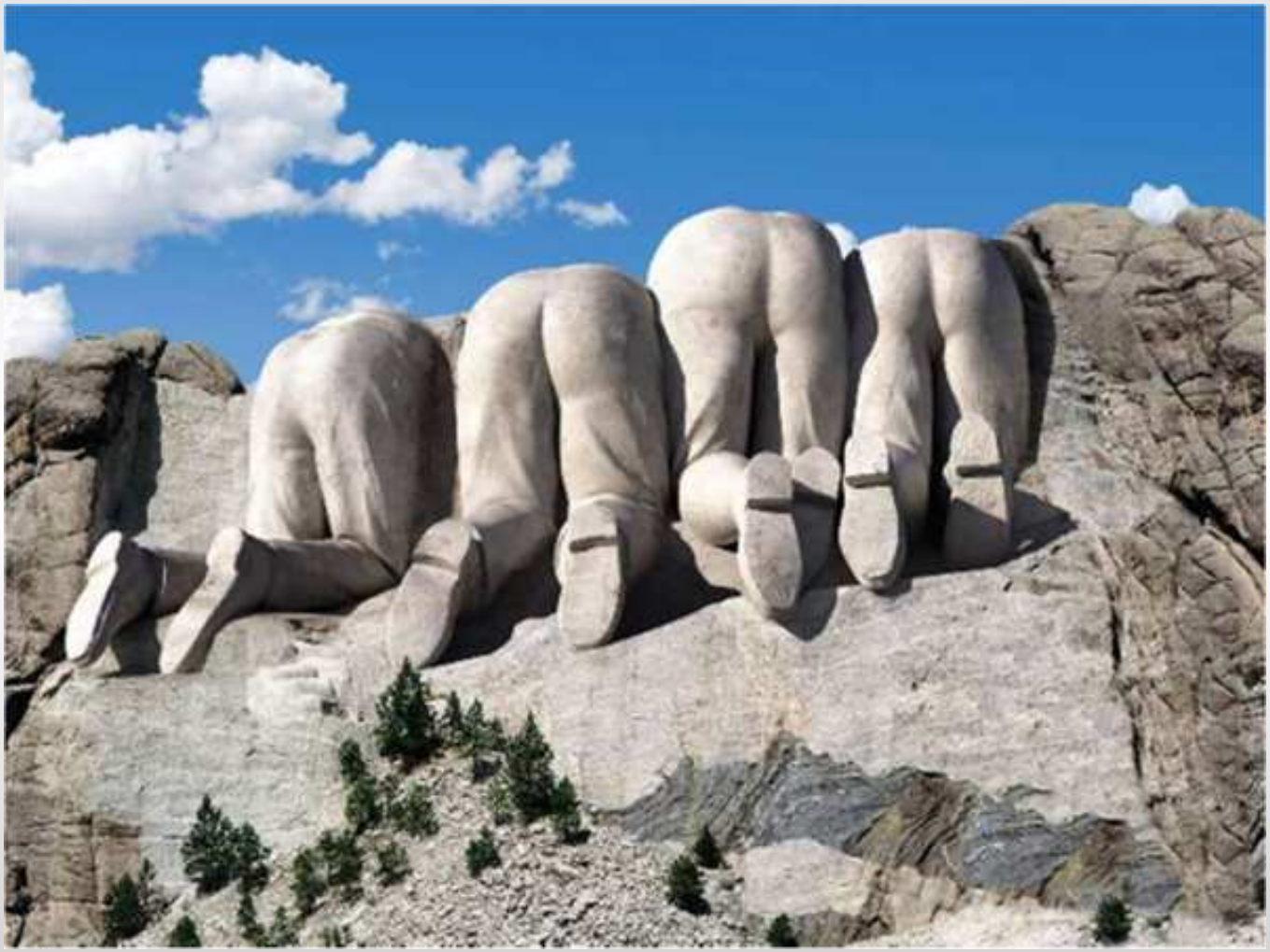
What is constraint folding

Soft Constraints allow for shifting points in the landscape up or down

Mount Rushmore Today

What is constraint folding

Soft Constraints allow for shifting points in the landscape up or down

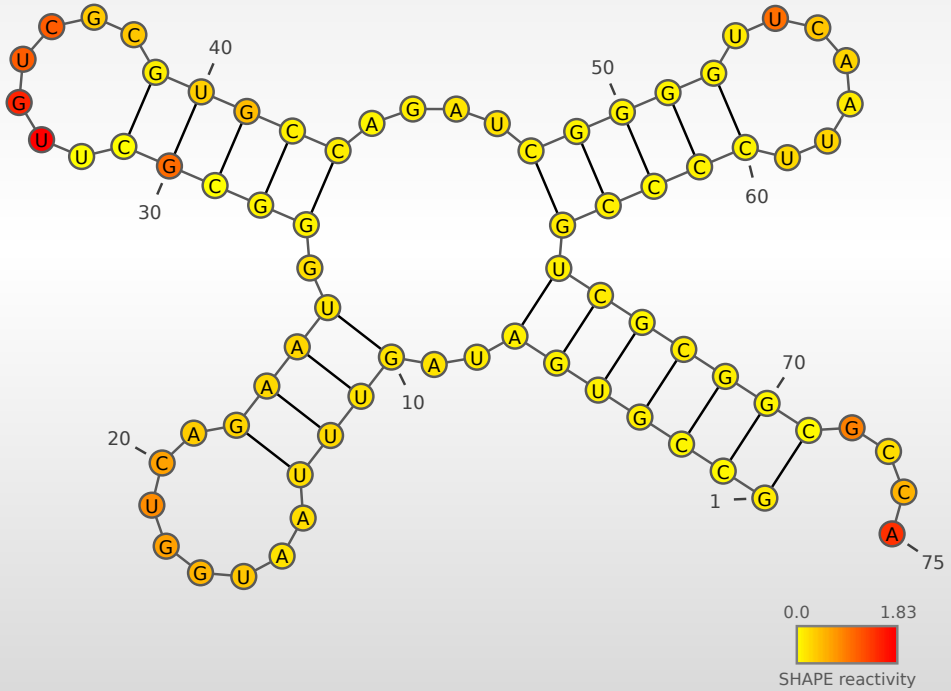


Mount Rushmore from the back

Soft constraints and SHAPE reactivity

SHAPE

- Selective 2’-hydroxyl acylation analyzed by primer extension
- Yields nucleotide flexibility
- Flexibility is inversely correlated to the pairing probability

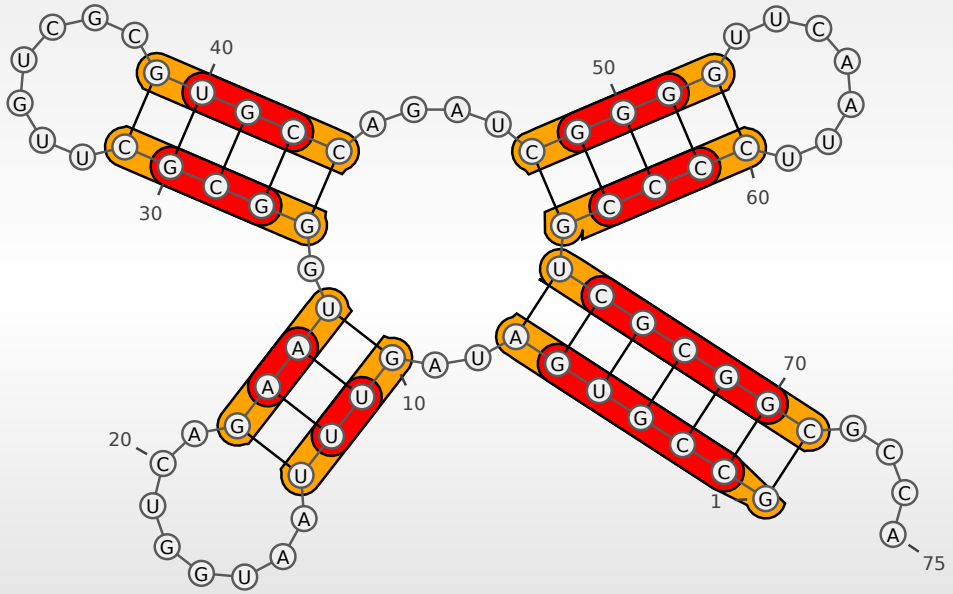


Soft constraints and SHAPE reactivity

Pseudo energy terms

- Degian et al. [2009] (stacked pairs)

$$\Delta\, G(i) = m * ln(reactivity[i] + 1) + b$$

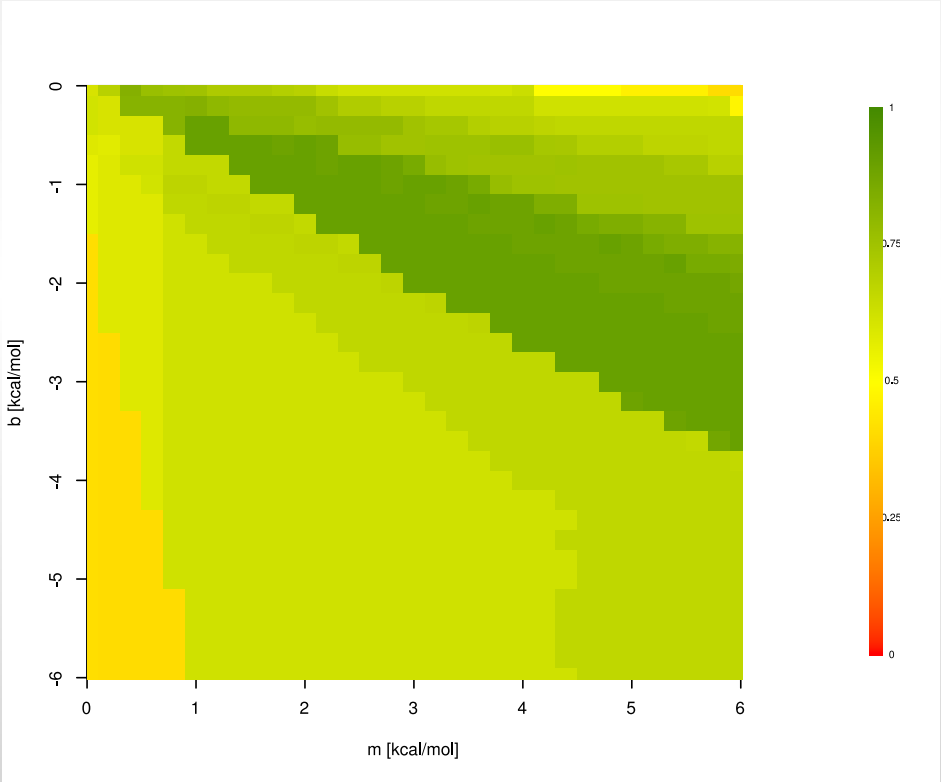


Soft constraints and SHAPE reactivity

Pseudo energy terms

- Degian et al. [2009] (stacked pairs)

$$\Delta\, G(i) = m * ln(reactivity[i] + 1) + b$$



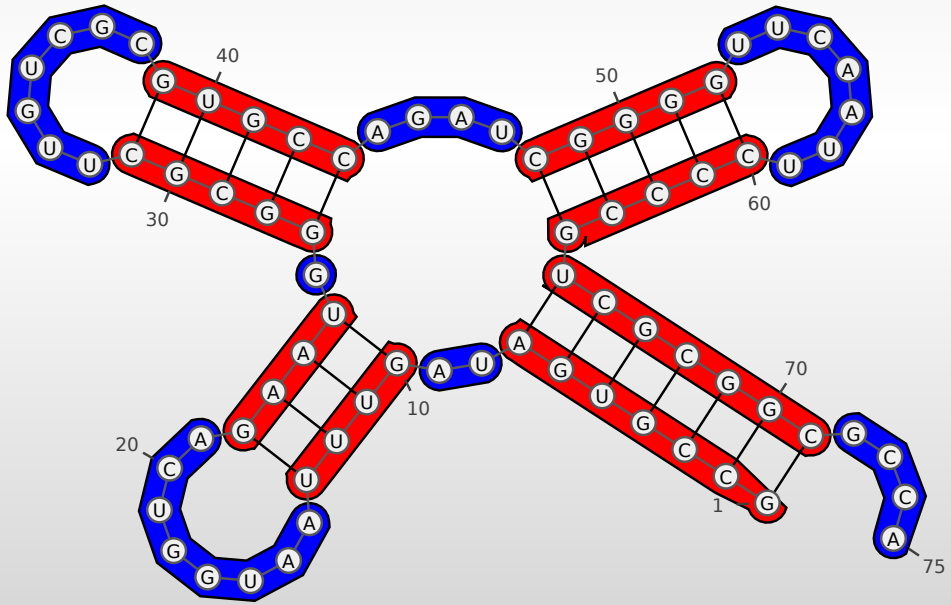
Soft constraints and SHAPE reactivity

Pseudo energy terms

- Zarringhalam et al. [2012] (unpaired bases and base pairs)

$$\Delta\ G(x,i) = \beta\ * |x - q_i|$$

$$x \in [0(\textit{unpaired}), 1(\textit{paired})]$$



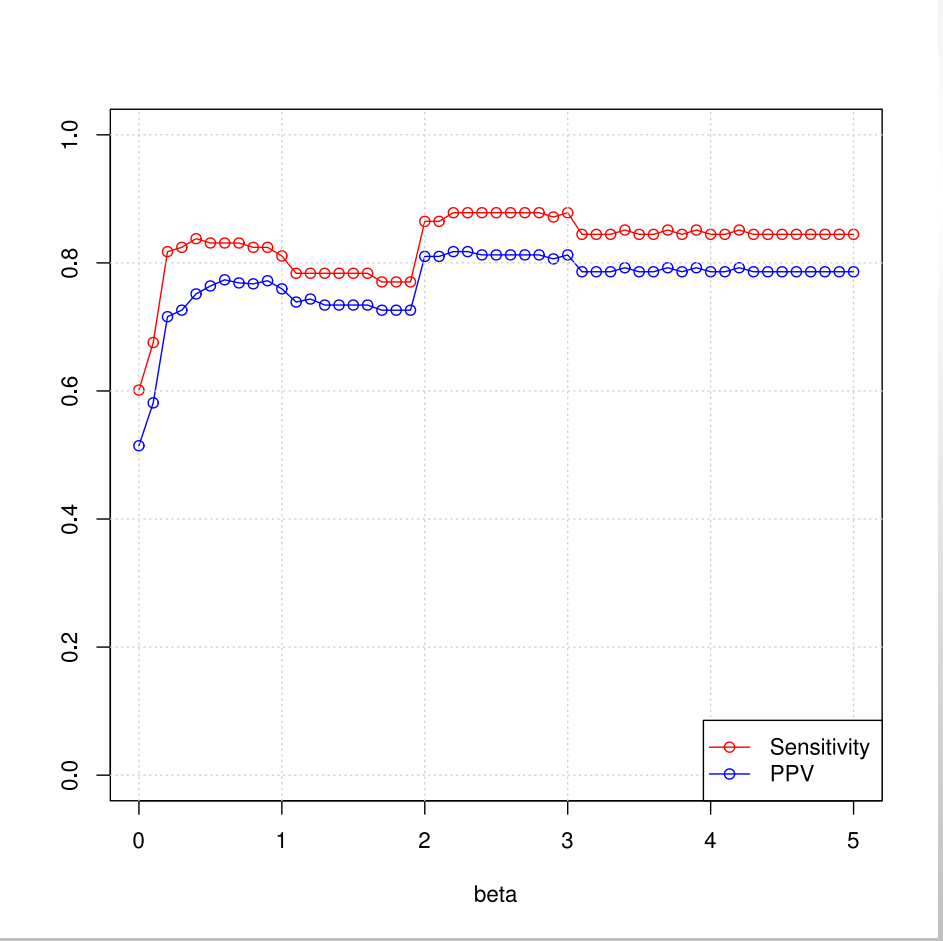
Soft constraints and SHAPE reactivity

Pseudo energy terms

- Zarringhalam et al. [2012] (unpaired bases and base pairs)

$$\Delta\, G(x,i) = \beta\ * |x - q_i|$$

$$x \in [0(\textit{unpaired}), 1(\textit{paired})]$$

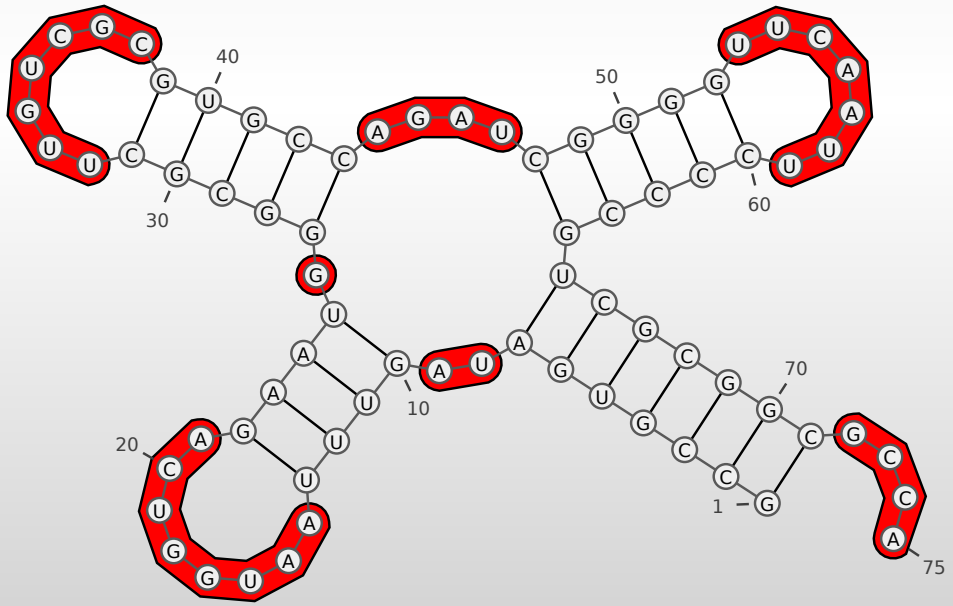


Soft constraints and SHAPE reactivity

Pseudo energy terms

- Washietl et al. [2012] (unpaired bases)
  Objective function

$$F(\vec{\epsilon}) = \sum_{i=1}^{n} \frac{\epsilon_i^2}{\tau^2} + \sum_{i=1}^{n} \frac{(p_i(\vec{\epsilon}) - q_i)^2}{\sigma^2} \rightarrow min$$

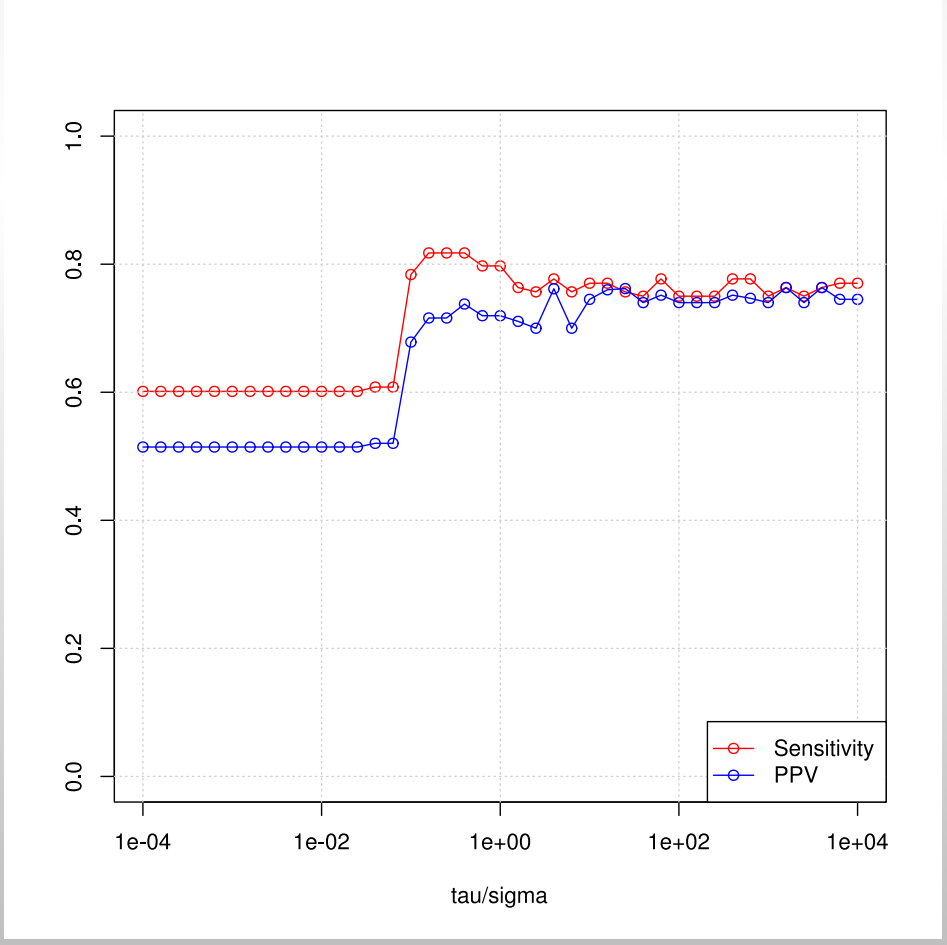


Soft constraints and SHAPE reactivity

Pseudo energy terms

- Washietl et al. [2012] (unpaired bases)
  Objective function

$$F(\vec{\epsilon}) = \sum_{i=1}^{n} \frac{\epsilon_i^2}{\tau^2} + \sum_{i=1}^{n} \frac{(p_i(\vec{\epsilon}) - q_i)^2}{\sigma^2} \rightarrow min$$



What is constraint folding

Secondary structure constraints aware programs:

- UNAfold[3] (hard)
- RNAstructure[4] (hard, soft)
- RNApbfold[5] (hard, soft)
- ViennaRNA Package v2.1.7[6] (hard)
- ViennaRNA Package v2.2[7] (hard, soft)

[3] Markham et al., 2008
[4] Reuter et al., 2010
[5] Washietl S. et al., 2012
[6] Hofacker et al., 1994, Lorenz et al. 2011
[7] not release yet

What is constraint folding - Examples

RNAstructure Hard constraints:

```
DS:
XA        # Nucleotides that will be double-stranded
-1
SS:
XB        # Nucleotides that will be single-stranded (unpaired)
-1
Mod:
XC        # Nucleotides accessible to chemical modification
-1
Pairs:
XD1 XD2   # Forced base pairs
-1 -1
FMN:
XE        # Nucleotides accessible to FMN cleavage
-1
Forbids:
XF1 XF2   # Prohibited base pairs
-1 -1
```

RNAstructure SHAPE support:

```
9    -999       # No reactivity information
10   -999
11   0.042816   # normalized SHAPE reactivity
12   0          # also a valid SHAPE reactivity
13   0.15027
14   0.16201
```

What is constraint folding - Examples

RNAfold Hard constraints (via pseudo dot-bracket string):

```
.   # no constraint for this base
|   # the corresponding base has to be paired
x   # the base is unpaired
<   # base i is paired with a base j>i
>   # base i is paired with a base j<i
( ) # base i pairs base j
```

Example:

```
$ cat test.seq
AUUUCCACUAGAGAAGGUCUAGAGUGUUUGUCGUUUGUCAGAAGUCCCUAUUCCAGGUAC
.....xxxx
$ RNAfold < test.seq
AUUUCCACUAGAGAAGGUCUAGAGUGUUUGUCGUUUGUCAGAAGUCCCUAUUCCAGGUAC
.....((((..(((...)))..))))((((.(....).)))).((.(((.....))).)) ( -7.50)
$ RNAfold -C < test.seq
WARNING: structure constraint is shorter than sequence
AUUUCCACUAGAGAAGGUCUAGAGUGUUUGUCGUUUGUCAGAAGUCCCUAUUCCAGGUAC
..........(((.(((....((.(.((((.(....).)))))..)))))..))).....  ( -5.10)
```

What is constraint folding

Where do current implementations apply structure constraints?

- positions that are unpaired
- base pair stacks
- base pairs

Are the above implementations sufficient?

What is constraint folding

Where do current implementations apply structure constraints?

- positions that are unpaired
- base pair stacks
- base pairs

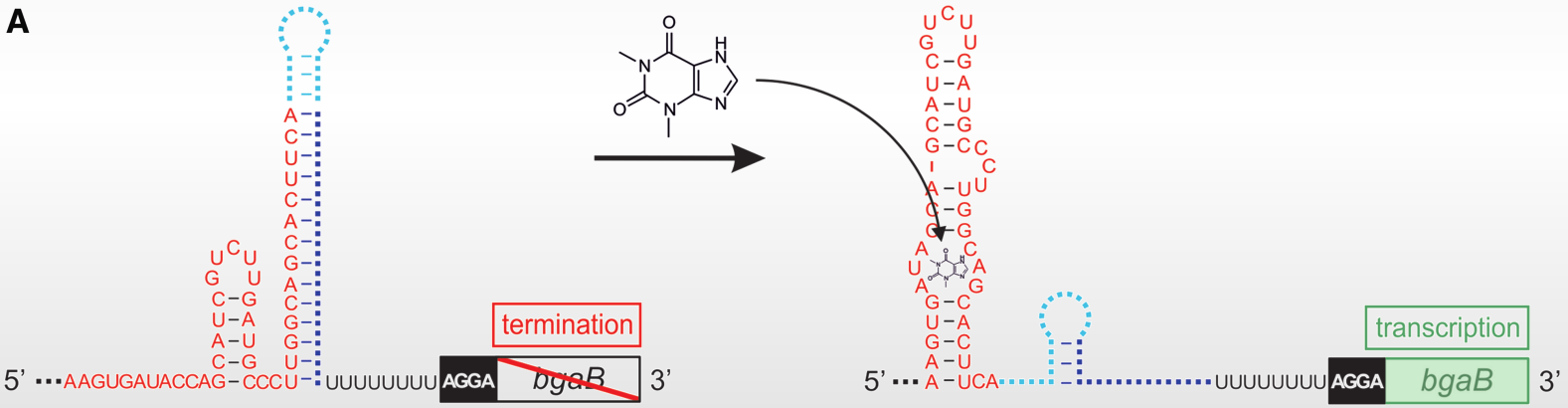
Are the above implementations sufficient?

Of course NOT!

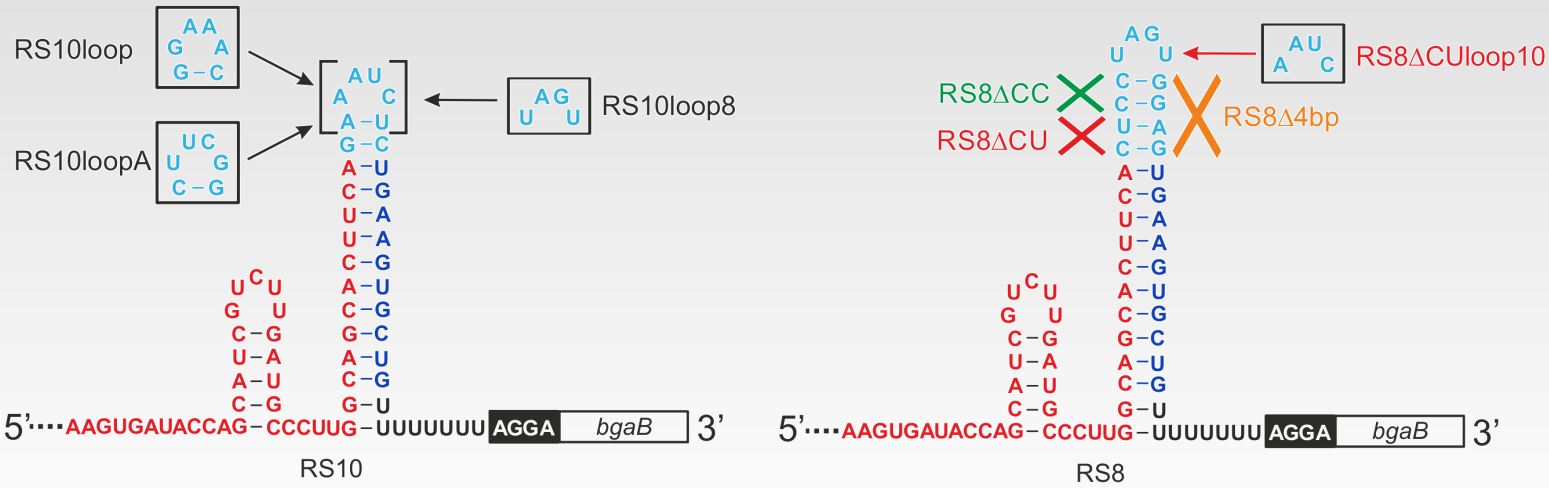
De novo design of theophylline sensing riboswitches

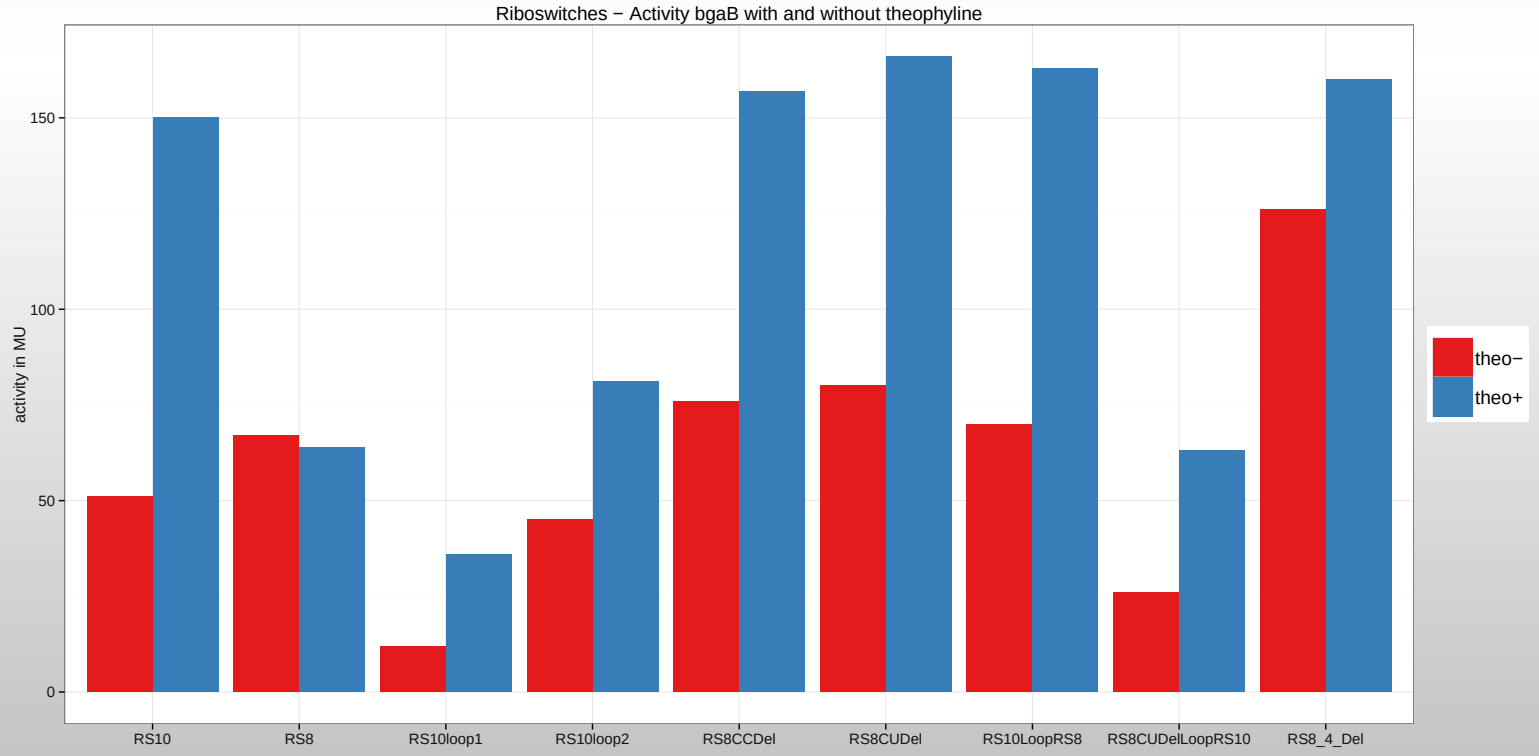
Wachsmuth et al. [2013]

- in silico design, in vivo validation (E. coli)
- theophylline aptamer upstream of a terminator hairpin
- aptamer fold overlaps with terminator
- ON switch upon presence of theophylline
- iterative design with `RNAinverse` and `RNAfold`



Mutational study





Bioinformatics analysis of the results

Cotranscriptional structure prediction and more:

- prediction of MFE and structure with `Cofold`[8]
- prediction of cotranscriptional folding with `kinwalker`[9]
- evaluation of free energies for 3bp-,4bp- and 5bp-seed of the terminators as a measure of how fast the terminator will form

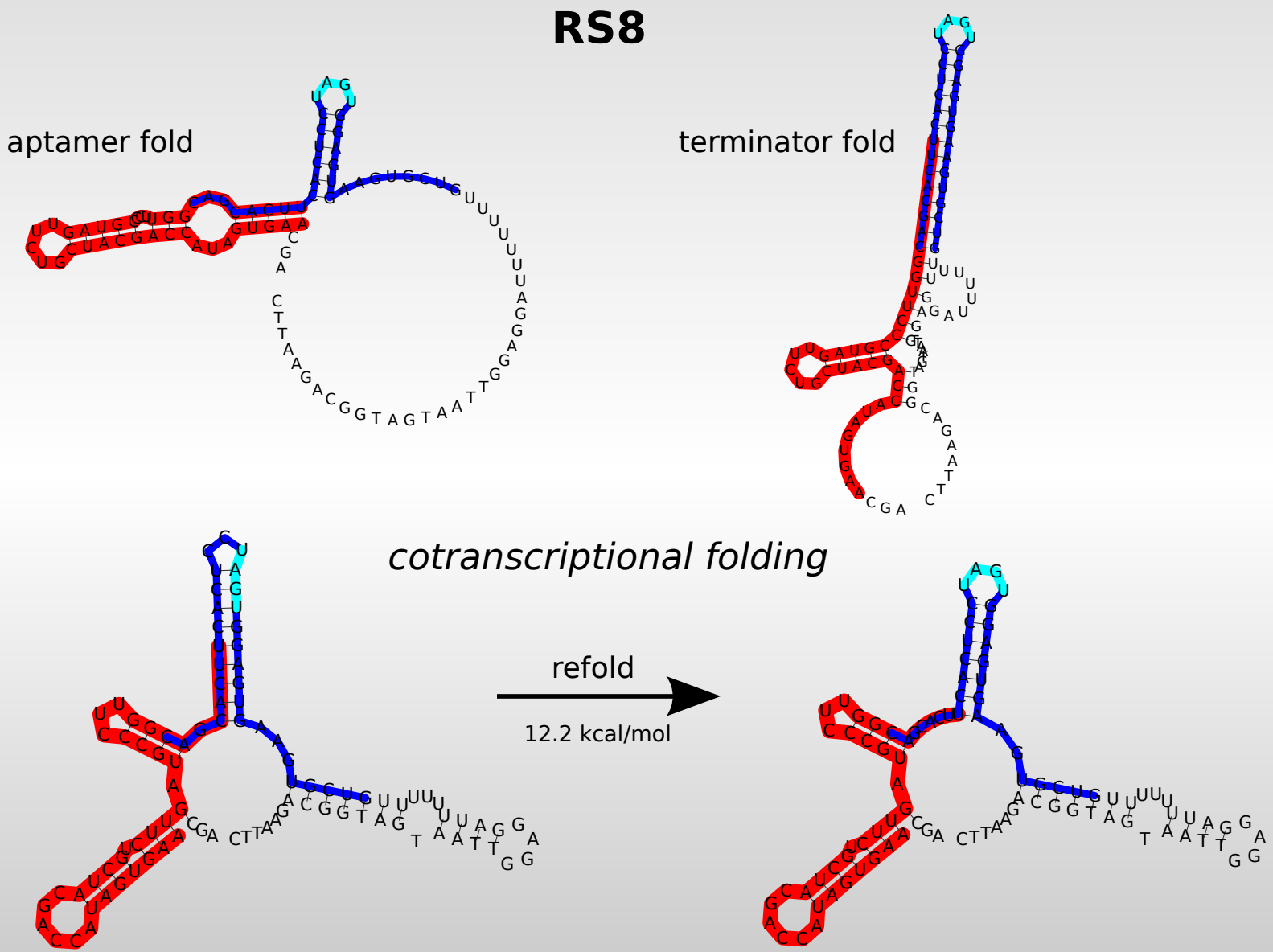
Results:

- `Cofold` output is the same as `RNAfold`
- `kinwalker` predicts cotranscriptional traps in 4 cases RS8, RS10loop2, RS8CCDel, RS8CUDel
- use terminator formation barrier as parameter

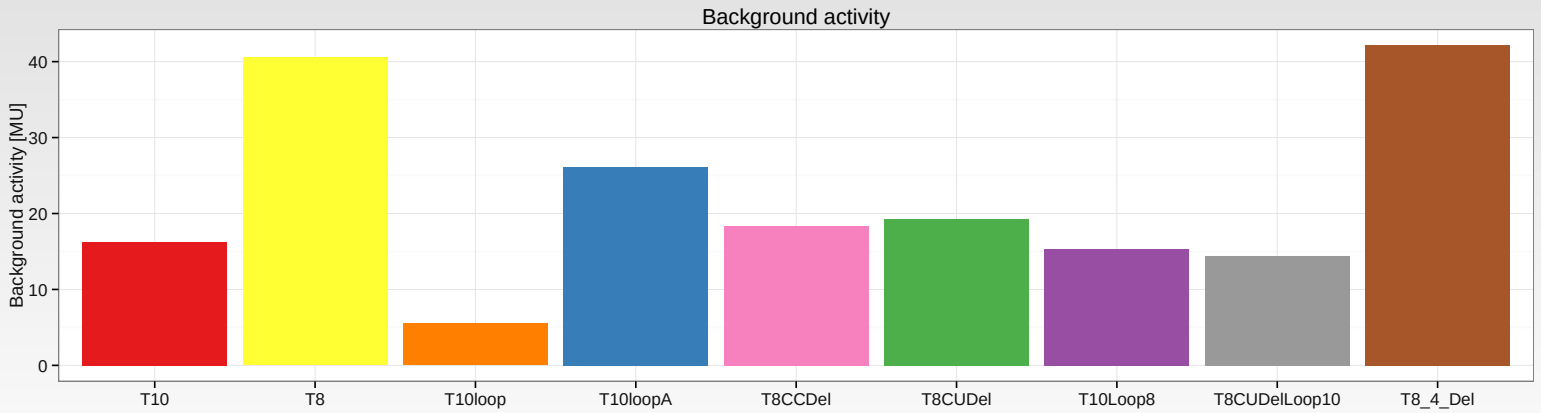
[8] Procter et al. 2013

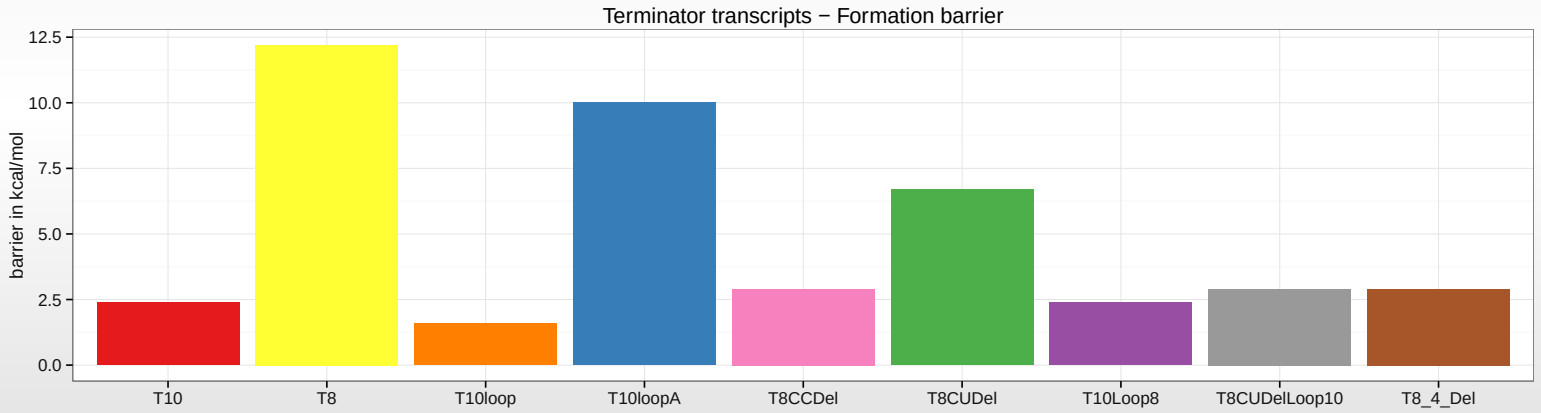
[9] Geis et al 2008

Hairpin formation barrier - RS8

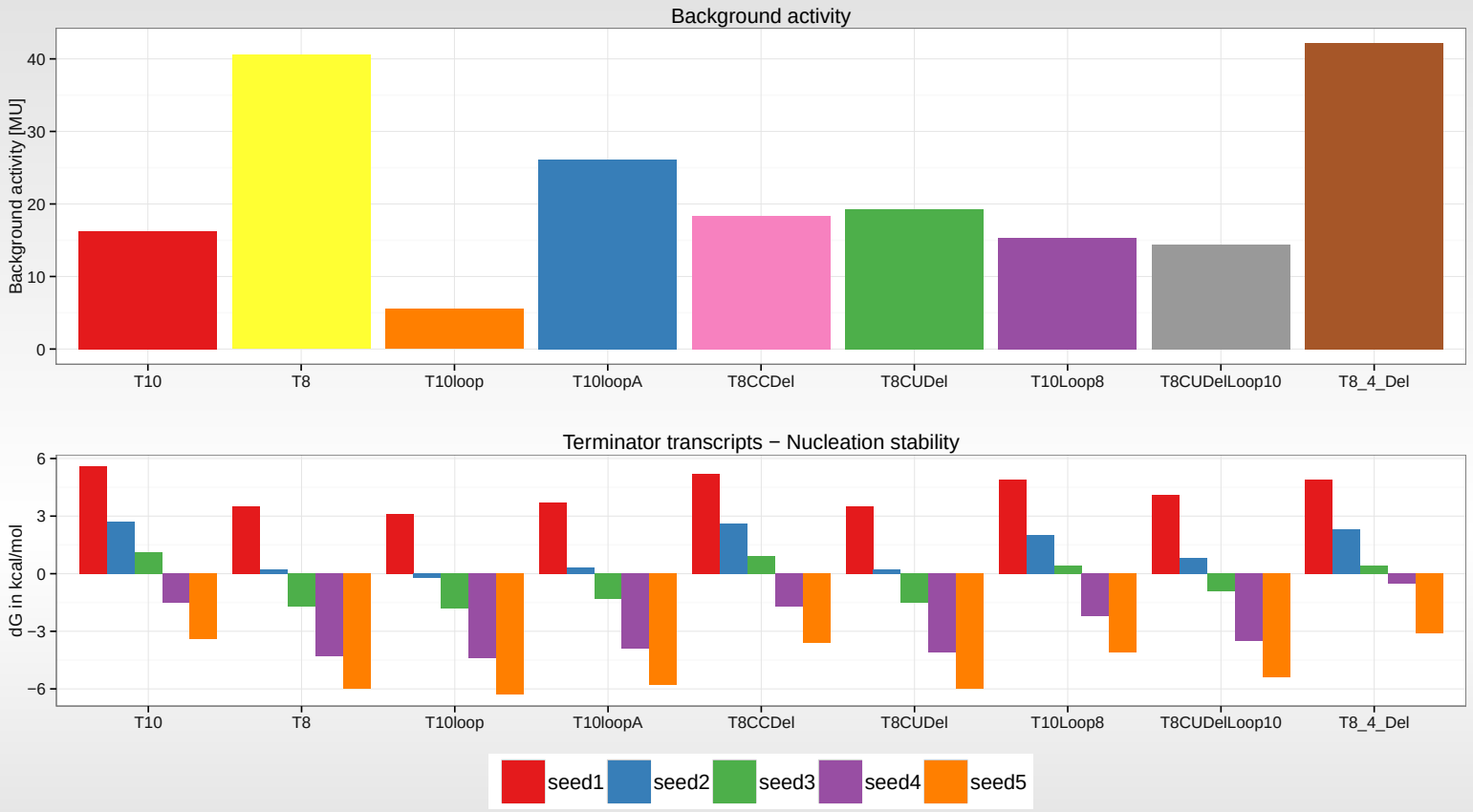


Hairpin formation barrier





Hairpin seed stabilities



Conclusion

- pure thermodynamic design is insufficient
- terminator seed performance seems to matter

  Best design: Tetra-loops (`GAAA`, `UUCG`) and strong closing pairs
- cotranscriptional effects have to be taken care of

  Design must exclude hairpins attenuating the terminator
- available terminator efficiency scores may be misleading
- construct new measure that incorporates the above parameters

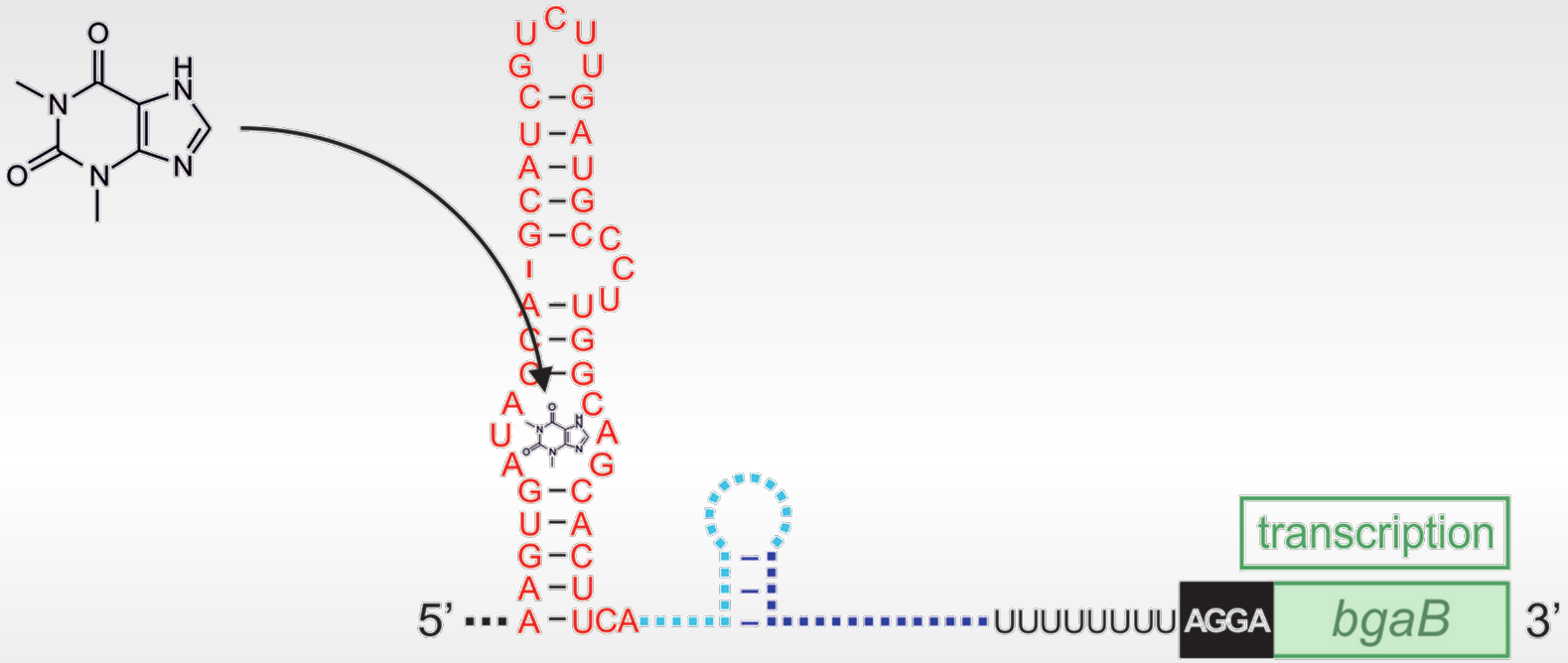
Conclusion

- pure thermodynamic design is insufficient
- terminator seed performance seems to matter

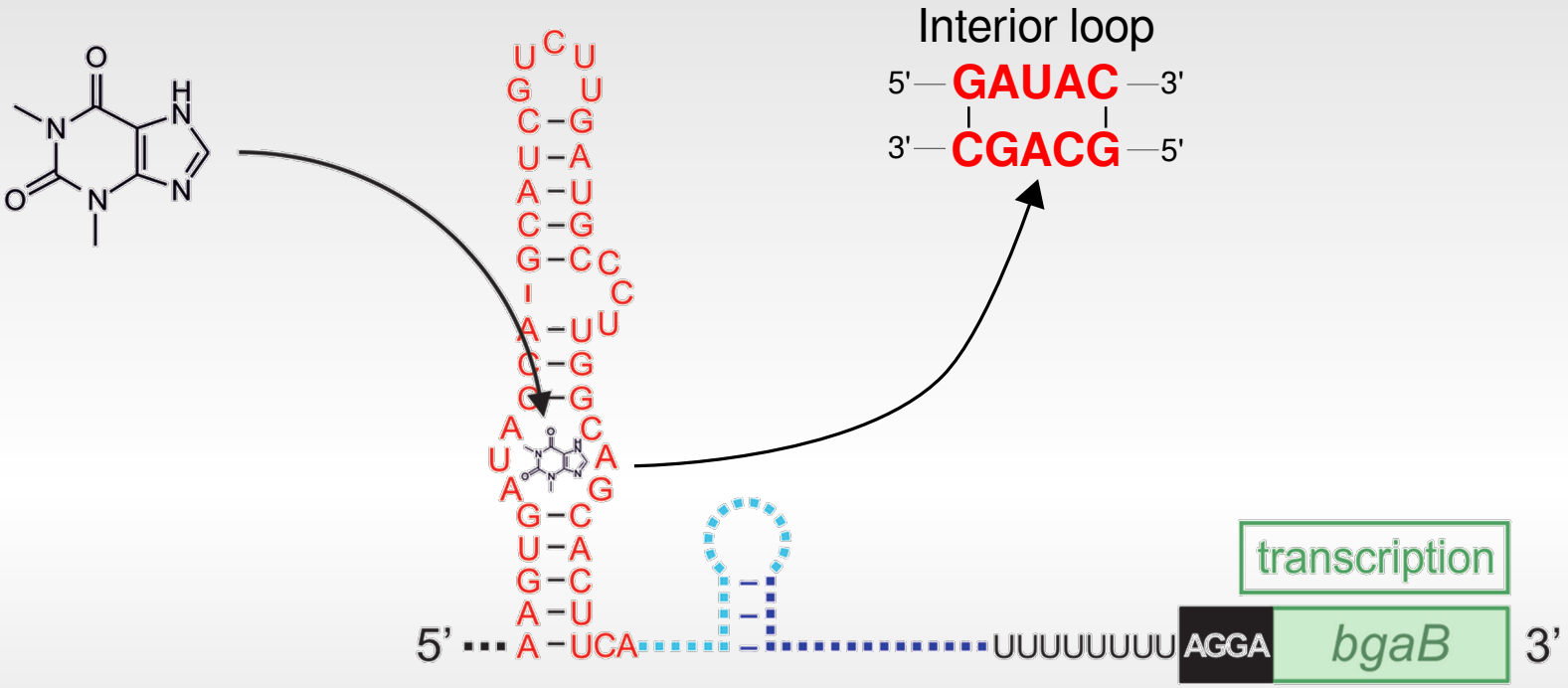
  Best design: Tetra-loops (`GAAA`, `UUCG`) and strong closing pairs
- cotranscriptional effects have to be taken care of

  Design must exclude hairpins attenuating the terminator
- available terminator efficiency scores may be misleading
- construct new measure that incorporates the above parameters
- What is the ligands influence on transcription and final fold?

Soft constraints for ligand - aptamer binding?!



Soft constraints for ligand - aptamer binding?!



Soft constraints for ligand - aptamer binding?!

Where do current implementations apply structure constraints?

- positions that are unpaired
- base pair stacks
- base pairs

Are the above implementations sufficient?

Of course NOT!

We need some generalization of Hard-, and Soft-Constraints!

Generalized Hard constraints

Discriminate between the decomposition steps (loop types)

1. Do something about base pairs:
   - (dis)allow particular base pair to appear in exterior-, hairpin-, interior-, multibranch-loops
   - distinguish between enclosing and enclosed base pairs
2. Do something with unpaired nucleotides:
   - specify whether or not a nucleotide may be unpaired in a distinguished loop type

Example: Base pair (i,j) has to pair but may only enclose a multiloop

Hard constraints can be expressed in terms of a boolean function

$$f(\vec{x}, d, data) = 0|1$$

with nucleotide position vector $\vec{x}$, decomposition step d and some *data* structure for, e.g. precalculated stuff.

Generalized Soft constraints

Generalization similar to Hard constraints

1. Discriminate between the decomposition steps
2. Generalize to a pseudo-energy function

$$f(\vec{x}, d, data) = e$$

to obtain bonus/malus for a particular decomposition step

Generalized Soft constraints

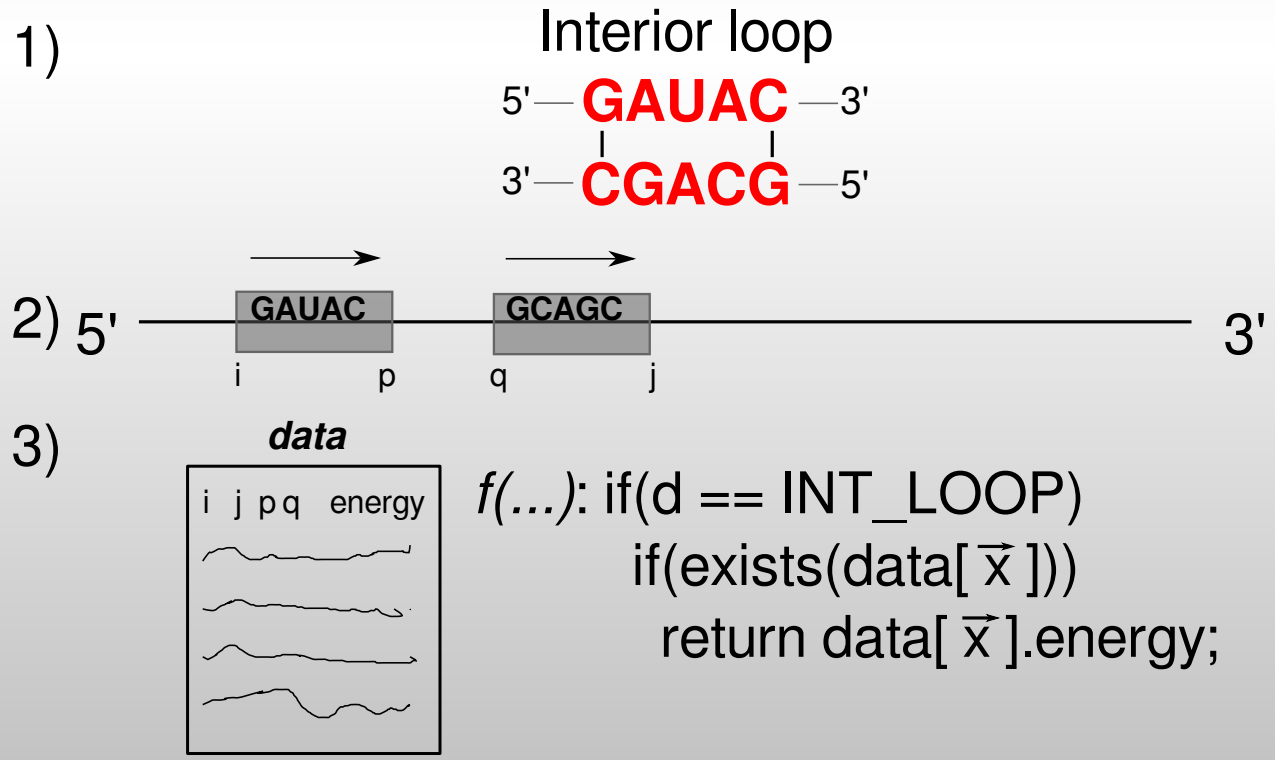
Generalization similar to Hard constraints

1. Discriminate between the decomposition steps
2. Generalize to a pseudo-energy function

$$f(\vec{x}, d, data) = e$$

to obtain bonus/malus for a particular decomposition step

Example:



Generalized Constraints

What can we do now?

- Include contribtions for some ligand binding, e.g. when it binds to interior loop pocket with specific motif
- Include 2.5D structure motifs [10]
- apply funny distortions to the energy landscape

[10] given they are enclosed by two canonical base pairs and reasonable free energy/pseudo energy is available

An example application for generalized soft constraints

RNA2Dfold

- classified dynamic programming approach
- computes MFE or partition function for a set of distance classes
- a distance class is the set of all structures whith a specified base pair distance to two initially chosen reference structures
  E.g. with reference structures s_1 and s_2, the distance class $(5, 17)$ is populated with all structures s that fulfill

$$d_{BP}(s, s_1) = 5 \wedge d_{BP}(s, s_2) = 17$$

- underlying algorithm is rather slow in terms of asymptotic time complexity $O(N^7)$ and consumes a lot of memory $O(N^4)$

An example application for generalized soft constraints

RNA2Dfold

- classified dynamic programming approach
- computes MFE or partition function for a set of distance classes
- a distance class is the set of all structures whith a specified base pair distance to two initially chosen reference structures
  E.g. with reference structures s_1 and s_2, the distance class $(5, 17)$ is populated with all structures s that fulfill

$$d_{BP}(s, s_1) = 5 \wedge d_{BP}(s, s_2) = 17$$

- underlying algorithm is rather slow in terms of asymptotic time complexity $O(N^7)$ and consumes a lot of memory $O(N^4)$
- However, it can be used for e.g. barrier heuristics, metastable state detection or even for RNA folding kinetics computations (see my talk last year)

An example application for generalized soft constraints

RNA2Dfold

- classified dynamic programming approach
- computes MFE or partition function for a set of distance classes
- a distance class is the set of all structures whith a specified base pair distance to two initially chosen reference structures
  E.g. with reference structures s_1 and s_2, the distance class $(5, 17)$ is populated with all structures s that fulfill

$$d_{BP}(s, s_1) = 5 \wedge d_{BP}(s, s_2) = 17$$

- underlying algorithm is rather slow in terms of asymptotic time complexity $O(N^7)$ and consumes a lot of memory $O(N^4)$
- However, it can be used for e.g. barrier heuristics, metastable state detection or even for RNA folding kinetics computations (see my talk last year)

Question: Can this be done more efficiently?

Distortion of the energy landscape

Idea: Approximation of the RNA2Dfold distance classes

- Sample structures from the whole Boltzmann ensemble $O(n^3)$
- classify each sample according to two chosen reference structures
- retrieve the MFE representative of the sampled distance classes
- compute partition function for each resulting distance classes

Drawback:

Sampling would only retrieve structure states from the lower portion of the energy landscape

Distortion of the energy landscape

Use generalized Soft constraints to favorize structures according to their distance to the chosen reference structures s_1 and s_2

$$
\begin{aligned}
Q &= \sum_s \exp^{-E(s)/RT} \\
Q^{distorted} &= \sum_s f(\mathcal{S}, s, s_1, s_2) \cdot \exp^{-E(s)/RT} \\
&= \sum_s x^{d_{BP}(s,s_1)} \cdot y^{d_{BP}(s,s_2)} \cdot \exp^{-E(s)/RT}
\end{aligned}
$$

In pseudo energy notations with $x = \exp^{-x'/RT}$ and $y = \exp^{-y'/RT}$

$$
Q^{distorted} = \sum_s \exp^{-(E(s)+x' \cdot d_{BP}(s,s_1)+y' \cdot d_{BP}(s,s_2))/RT}
$$

Distortion the energy landscape

Now, choose x and y such that s_1 and s_2 and the MFE structure s_{MFE} are equally probable.

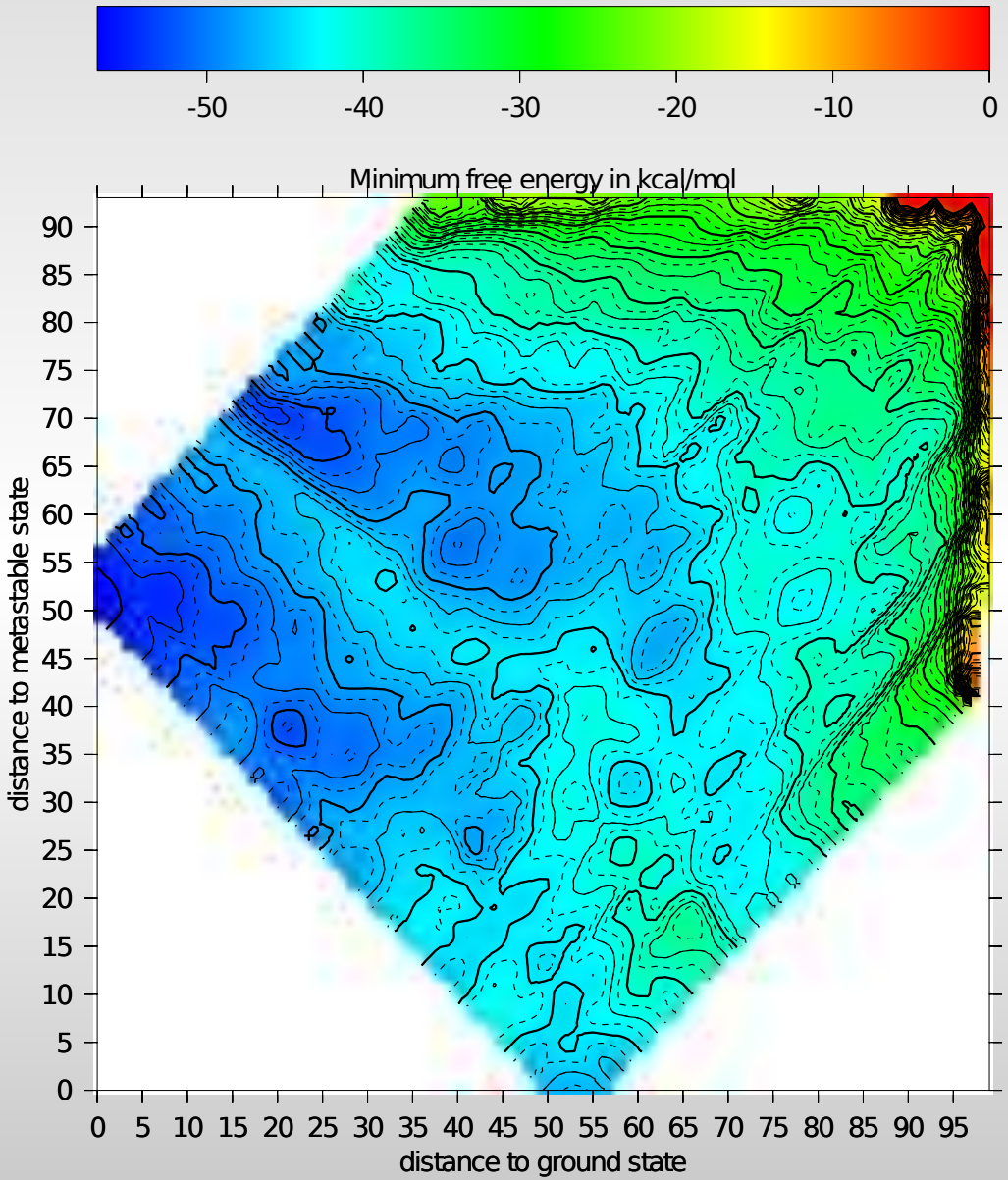
$$P(s) = \frac{\exp^{-E(s)/RT}}{Q}$$

$$\begin{aligned}
\exp^{-E(s_1)/RT} \cdot x^0 \cdot y^{d_{BP}(s_1,s_2)} &= \exp^{-E(s_2)/RT} \cdot x^{d_{BP}(s_1,s_2)} \cdot y^0 \\
\exp^{-E(s_1)/RT} \cdot x^0 \cdot y^{d_{BP}(s_1,s_2)} &= \exp^{-E(s_{MFE})/RT} \cdot x^{d_{BP}(s_1,s_{MFE})} \cdot y^{d_{BP}(s_2,s_{MFE})} \\
\exp^{-E(s_2)/RT} \cdot x^{d_{BP}(s_1,s_2)} \cdot y^0 &= \exp^{-E(s_{MFE})/RT} \cdot x^{d_{BP}(s_1,s_{MFE})} \cdot y^{d_{BP}(s_2,s_{MFE})}
\end{aligned}$$

The above equations can now be solved for x and y

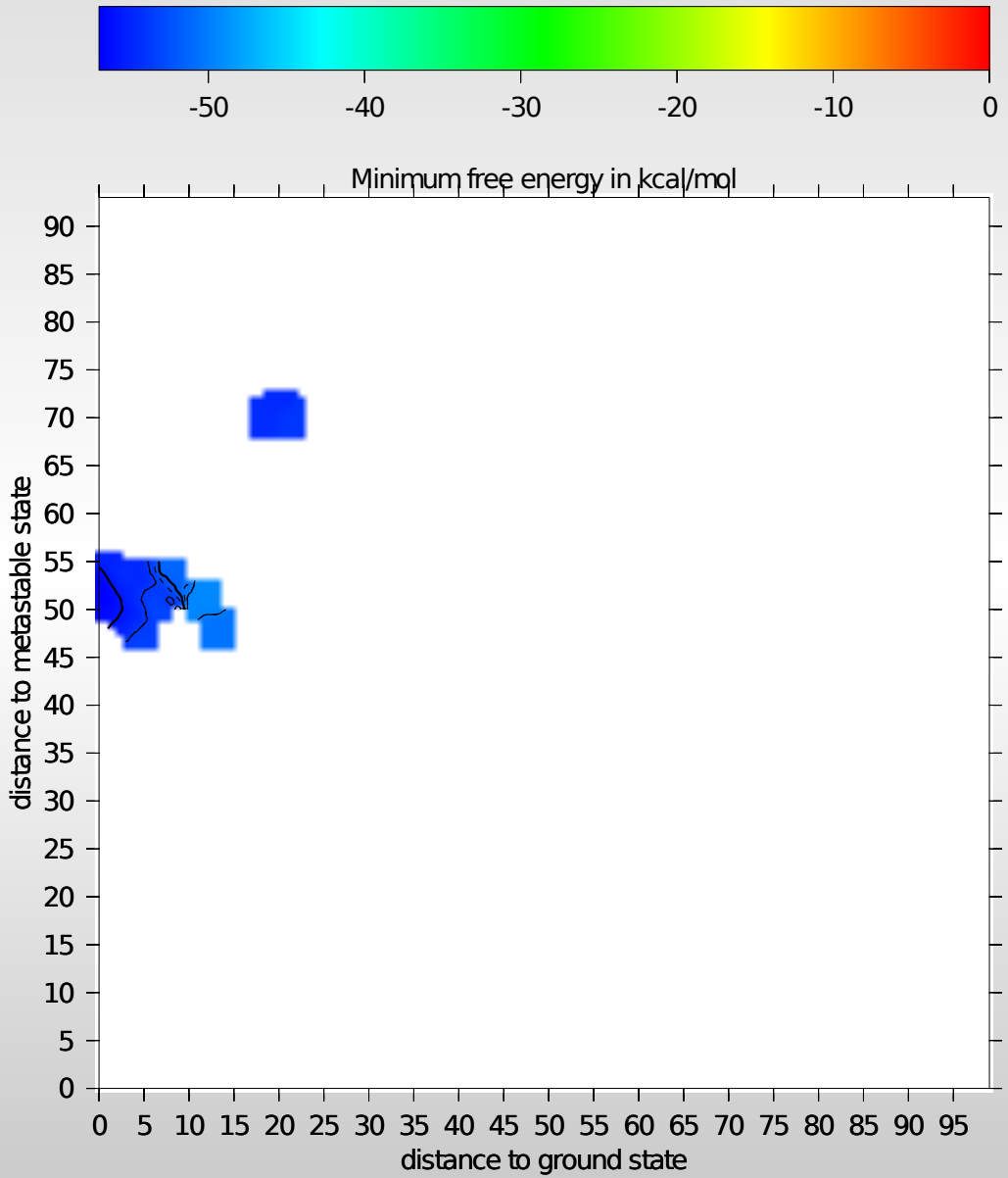
And then sample from this distorted landscape

Example 3: 5'-UTR in MS2[11] - RNA2Dfold

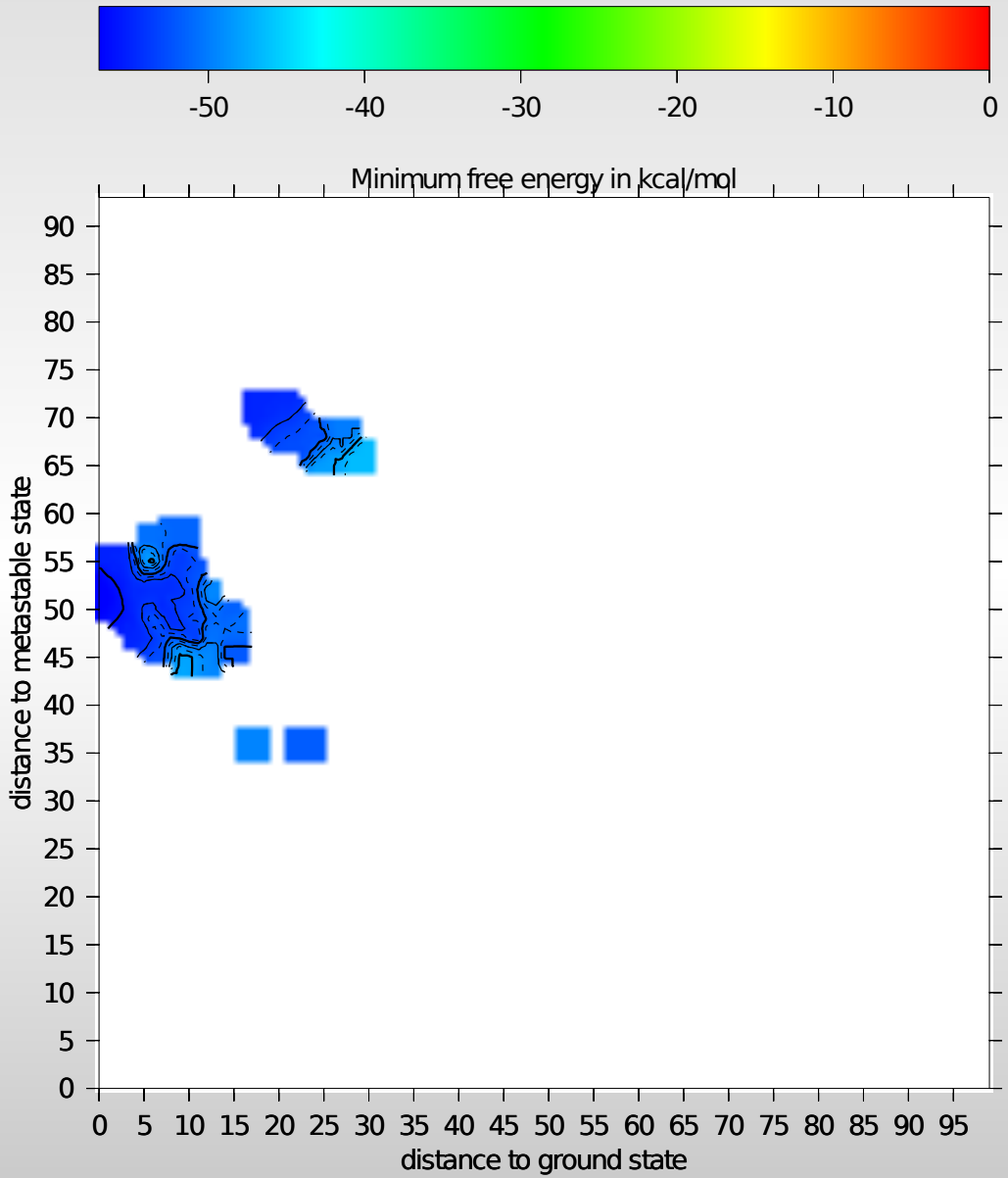


[11] van Meerten et al. 2001

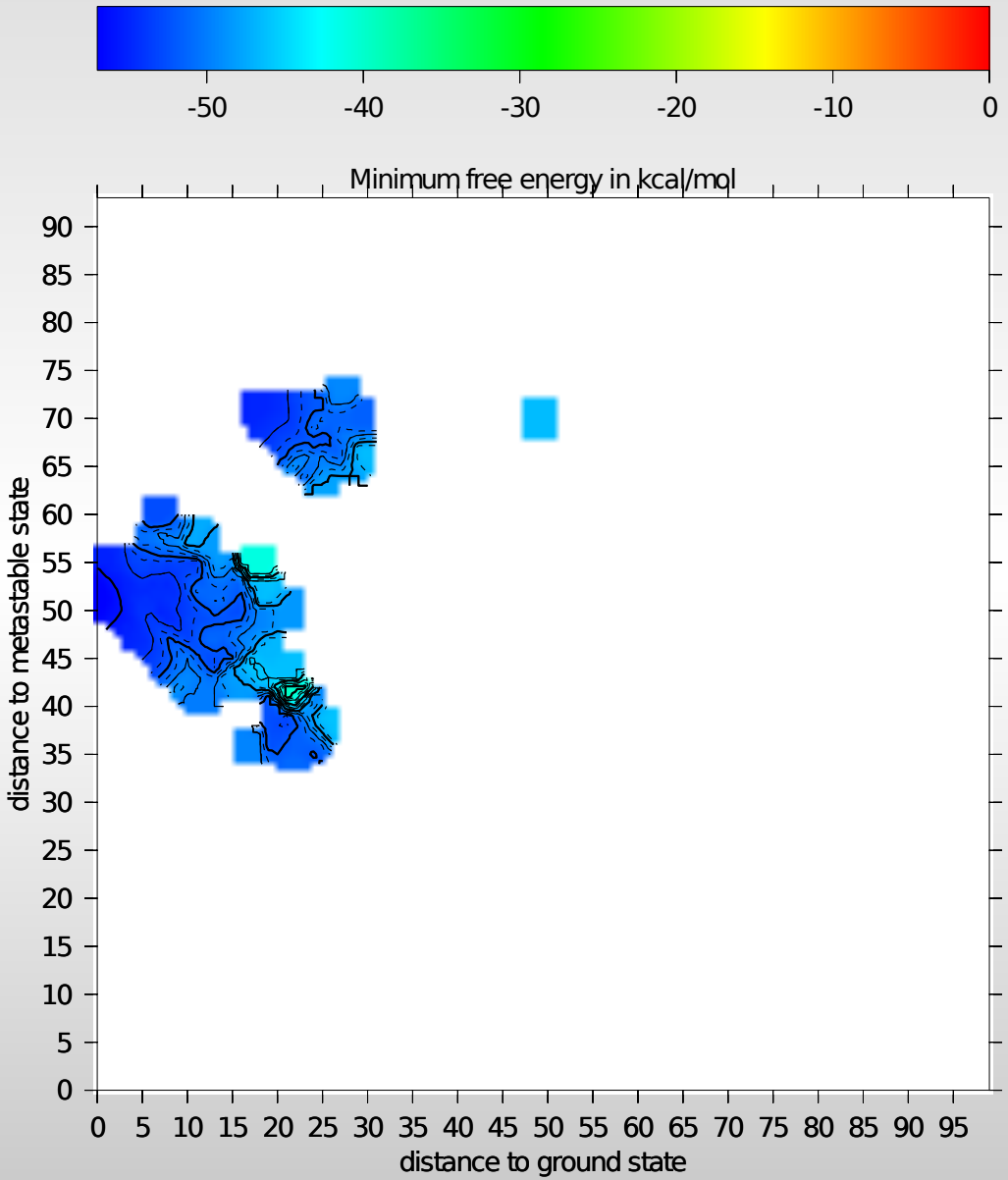
Example 3: 5'-UTR in MS2 - RNAsubopt -p 10^2



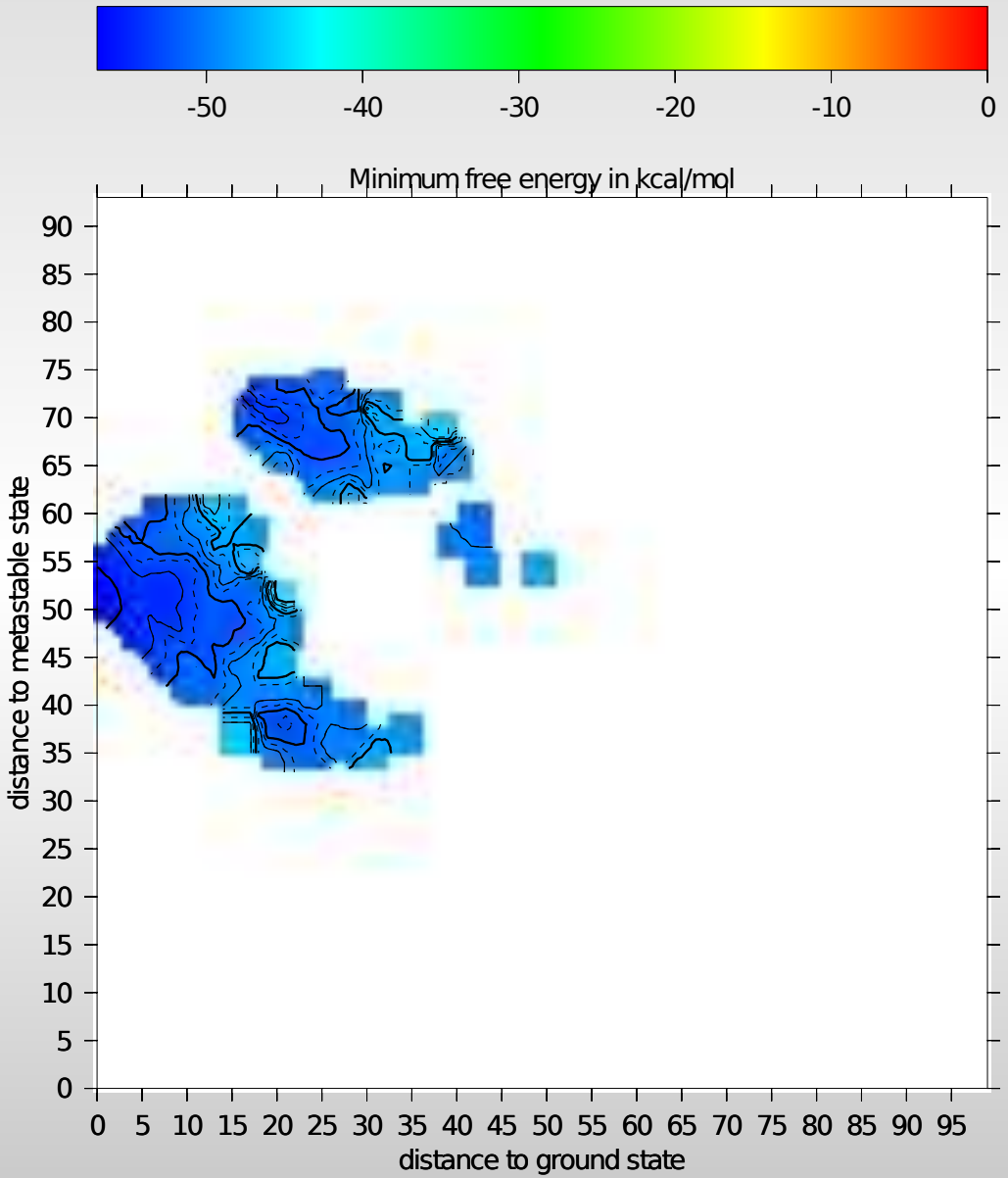
Example 3: 5'-UTR in MS2 - RNAsubopt -p 10^3



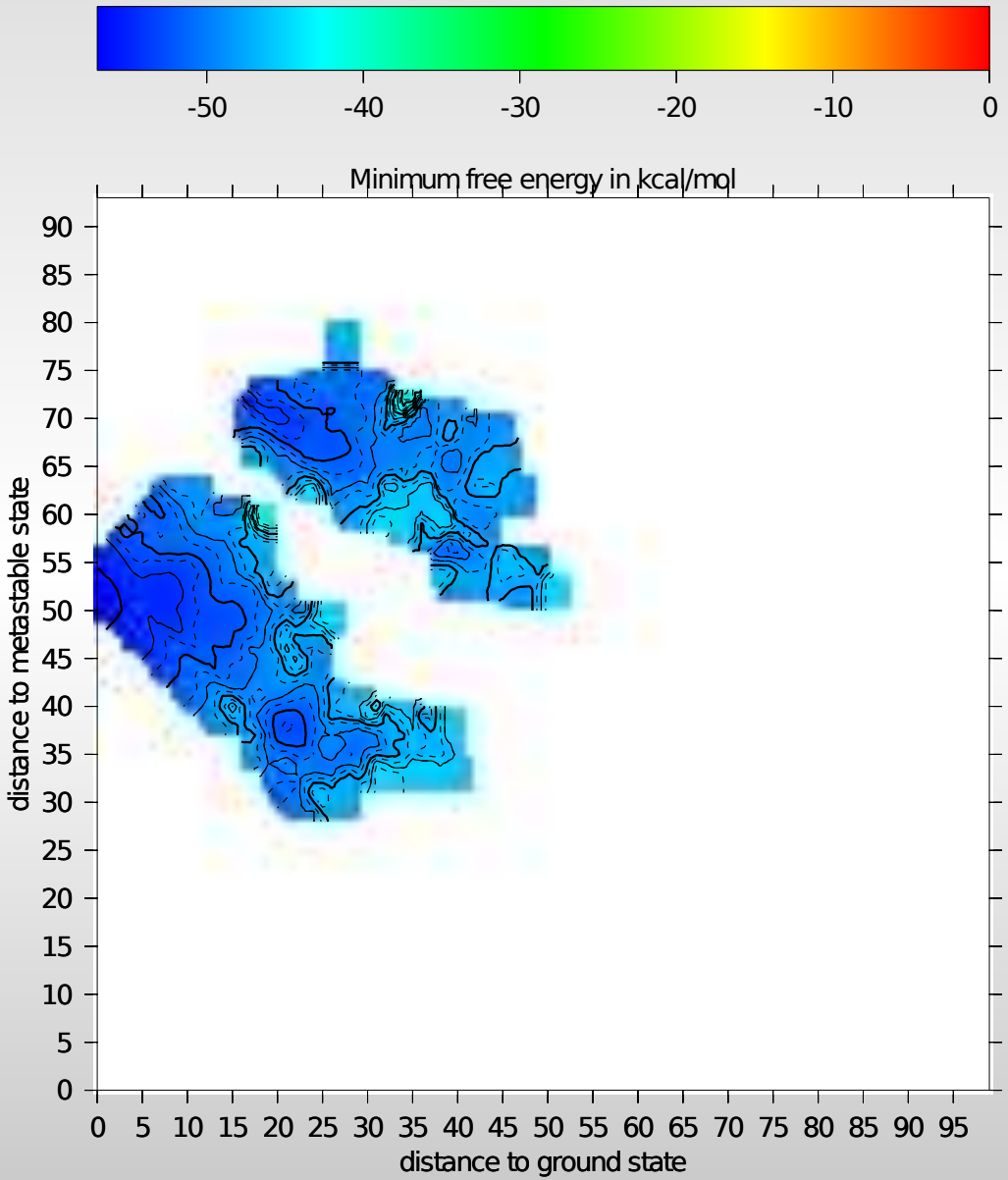
Example 3: 5'-UTR in MS2 - RNAsubopt -p 10^4



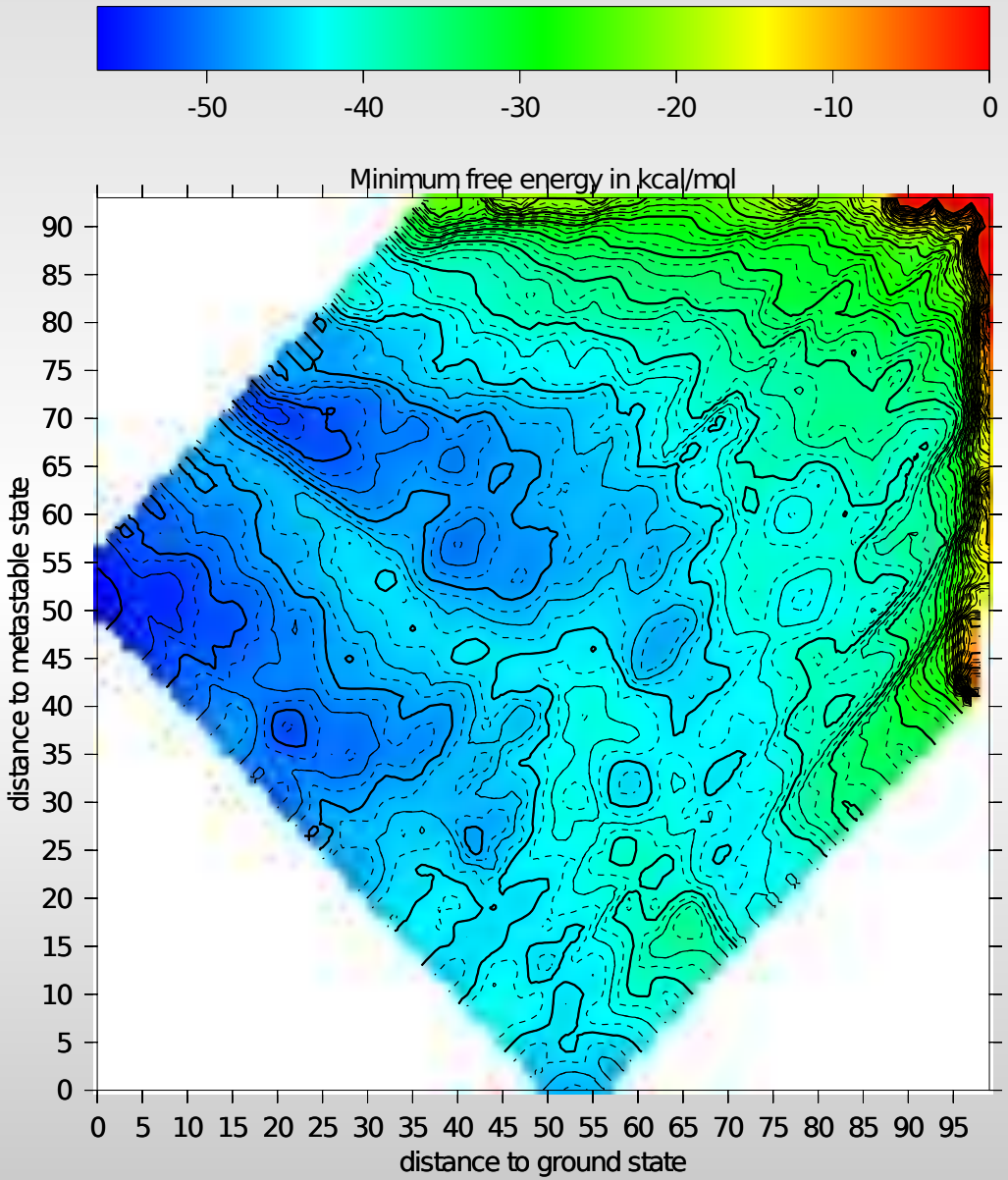
Example 3: 5'-UTR in MS2 - RNAsubopt -p 10^5



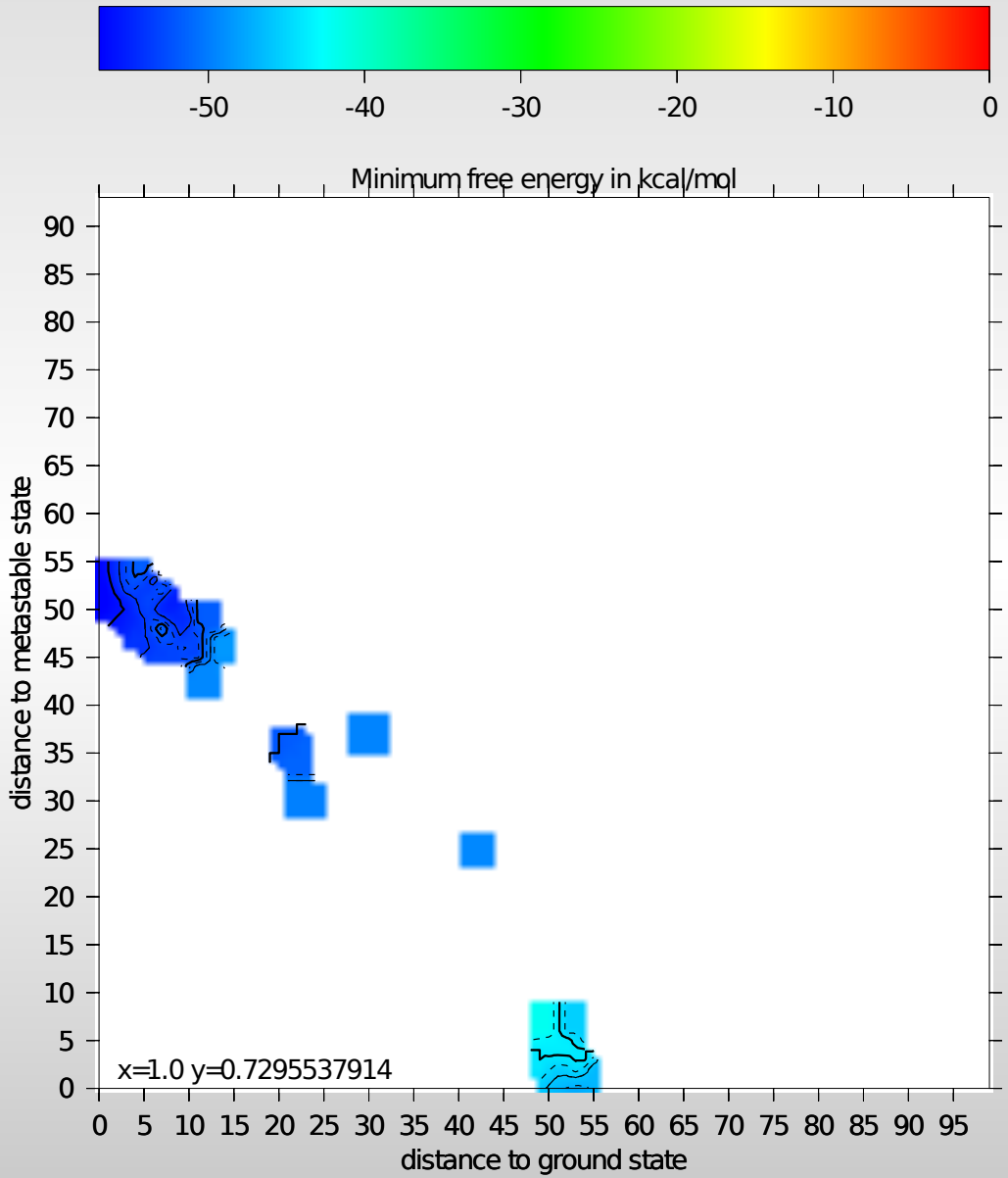
Example 3: 5'-UTR in MS2 - RNAsubopt -p 10^6



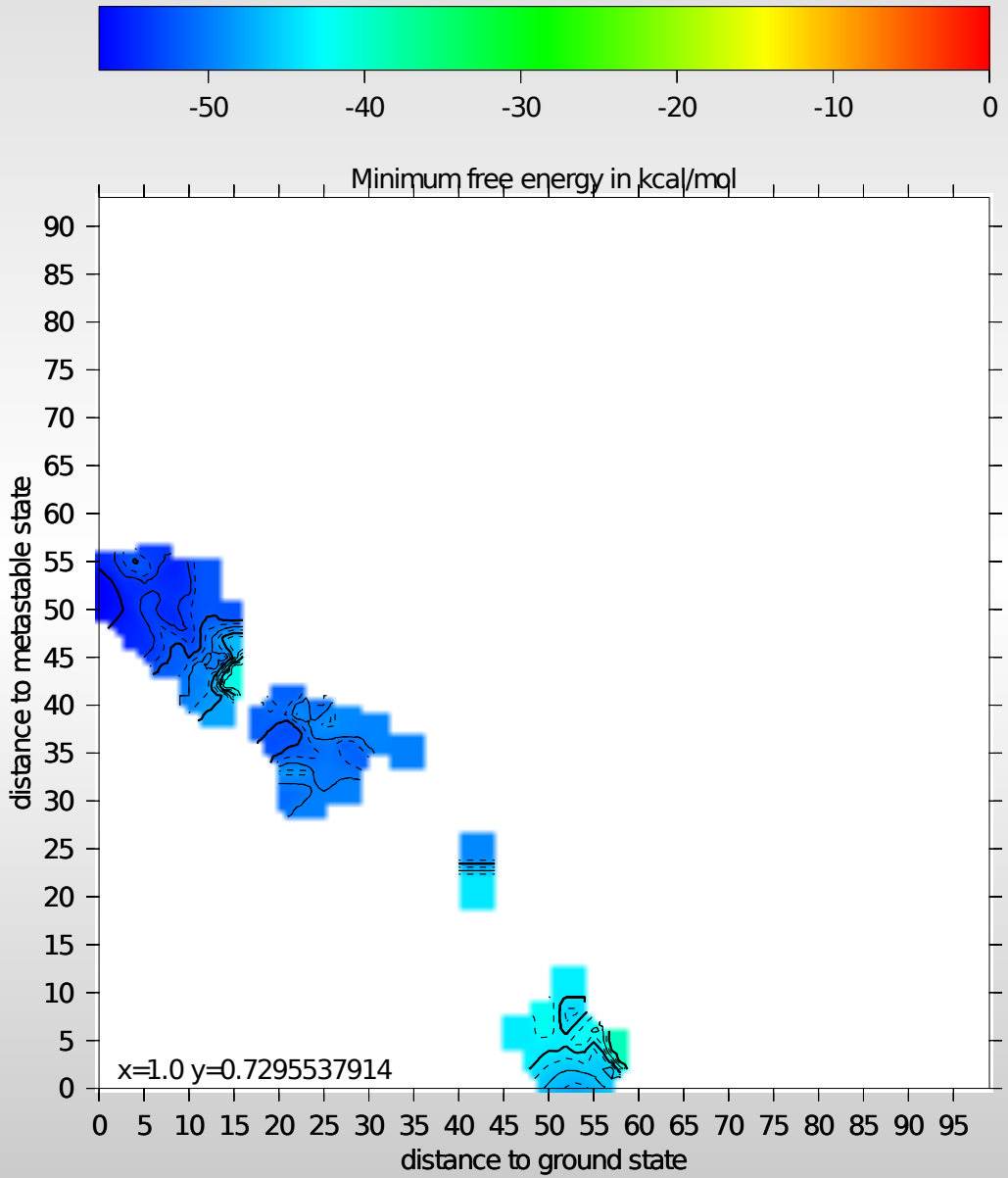
Example 3: 5'-UTR in MS2 - RNA2Dfold



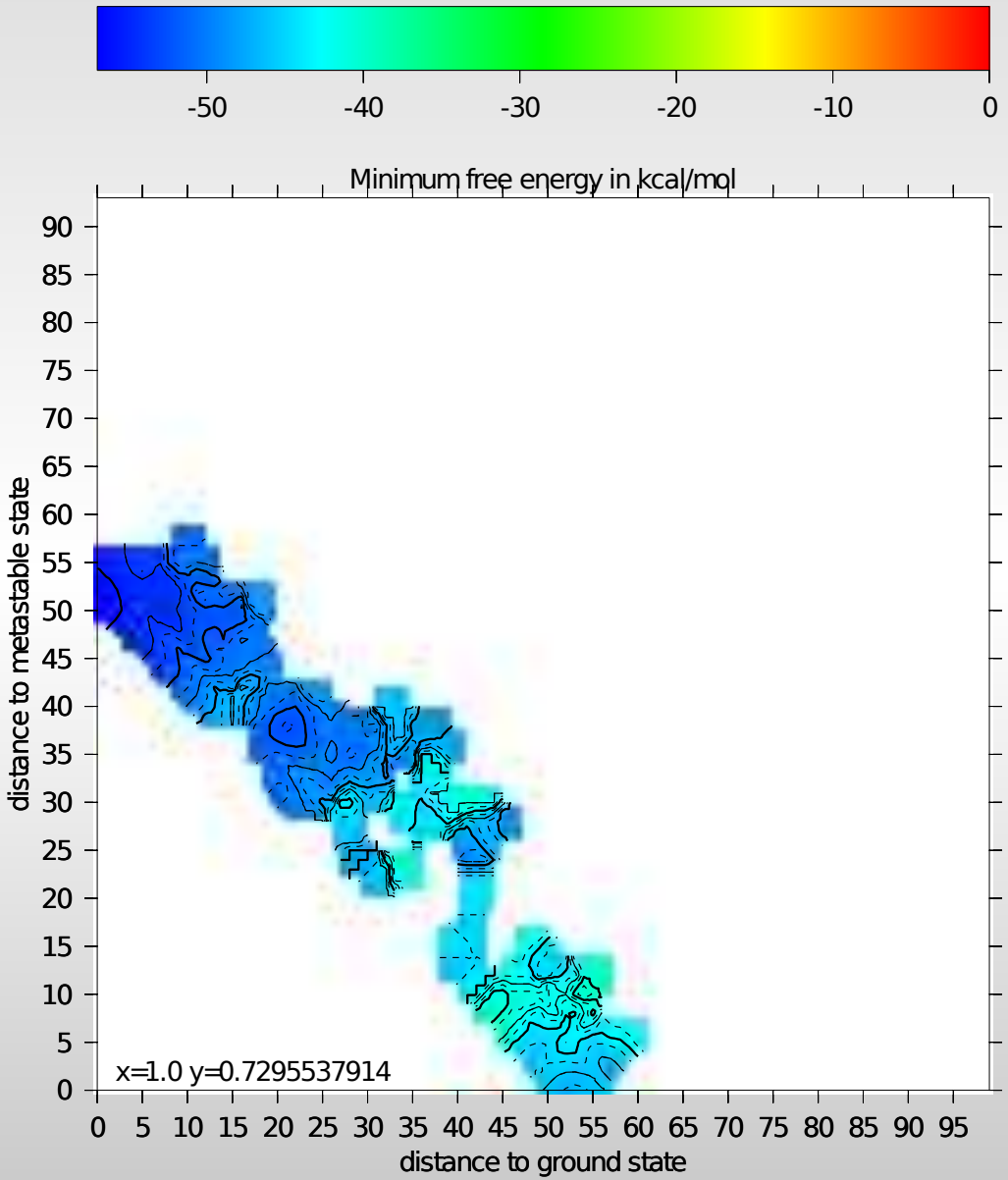
Example 3: 5'-UTR in MS2 - distortion 10^2



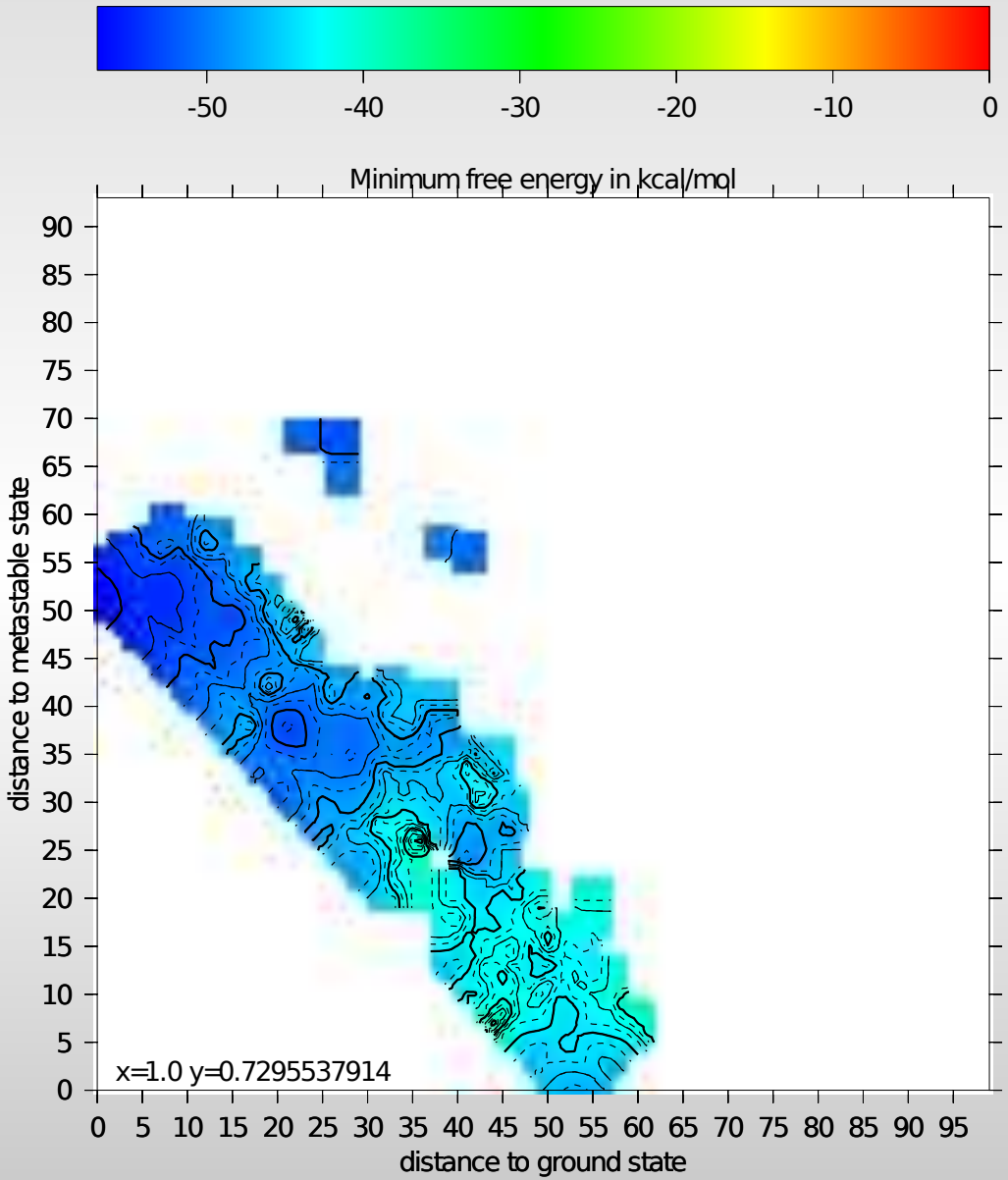
Example 3: 5'-UTR in MS2 - distortion 10^3



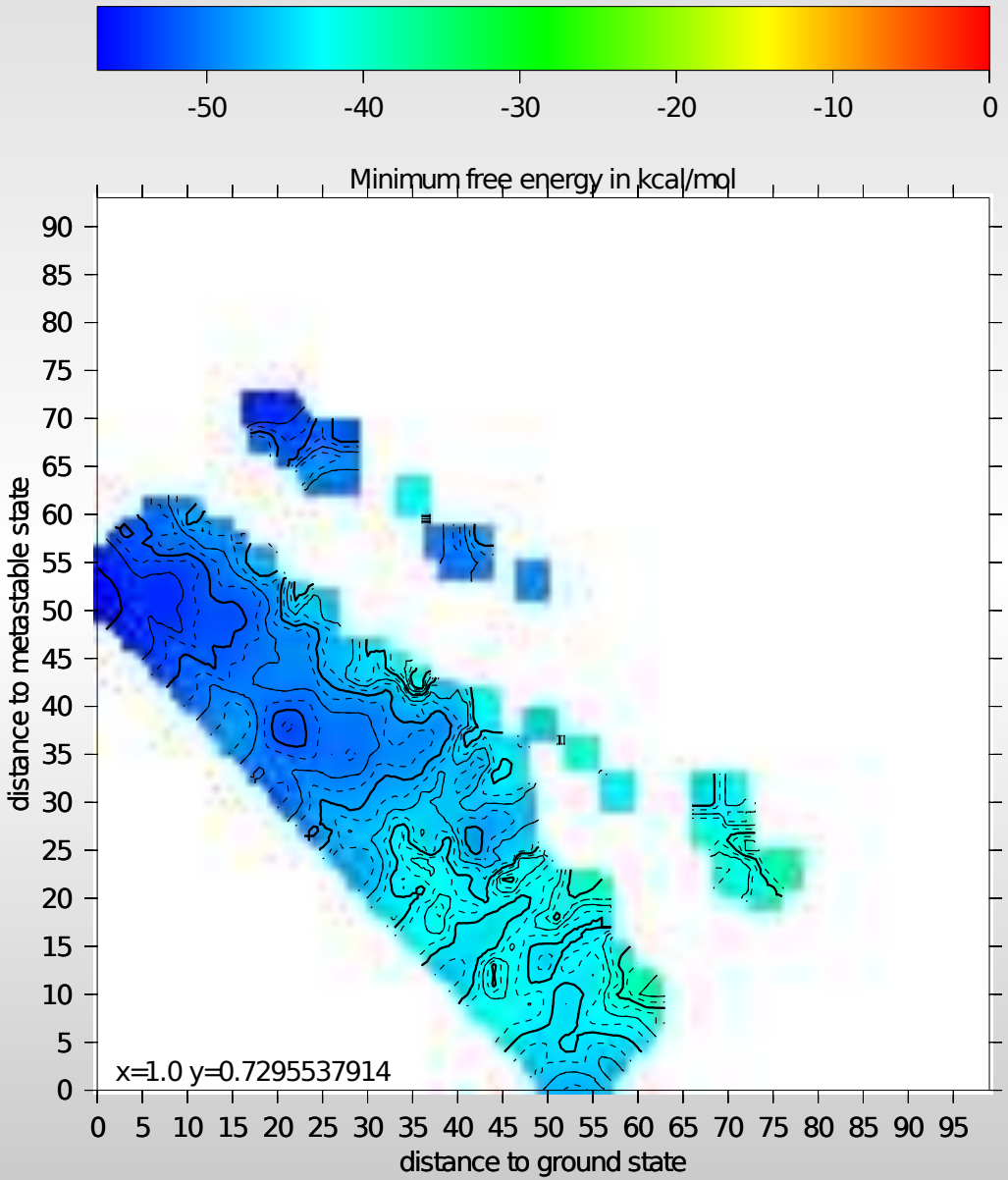
Example 3: 5'-UTR in MS2 - distortion 10^4



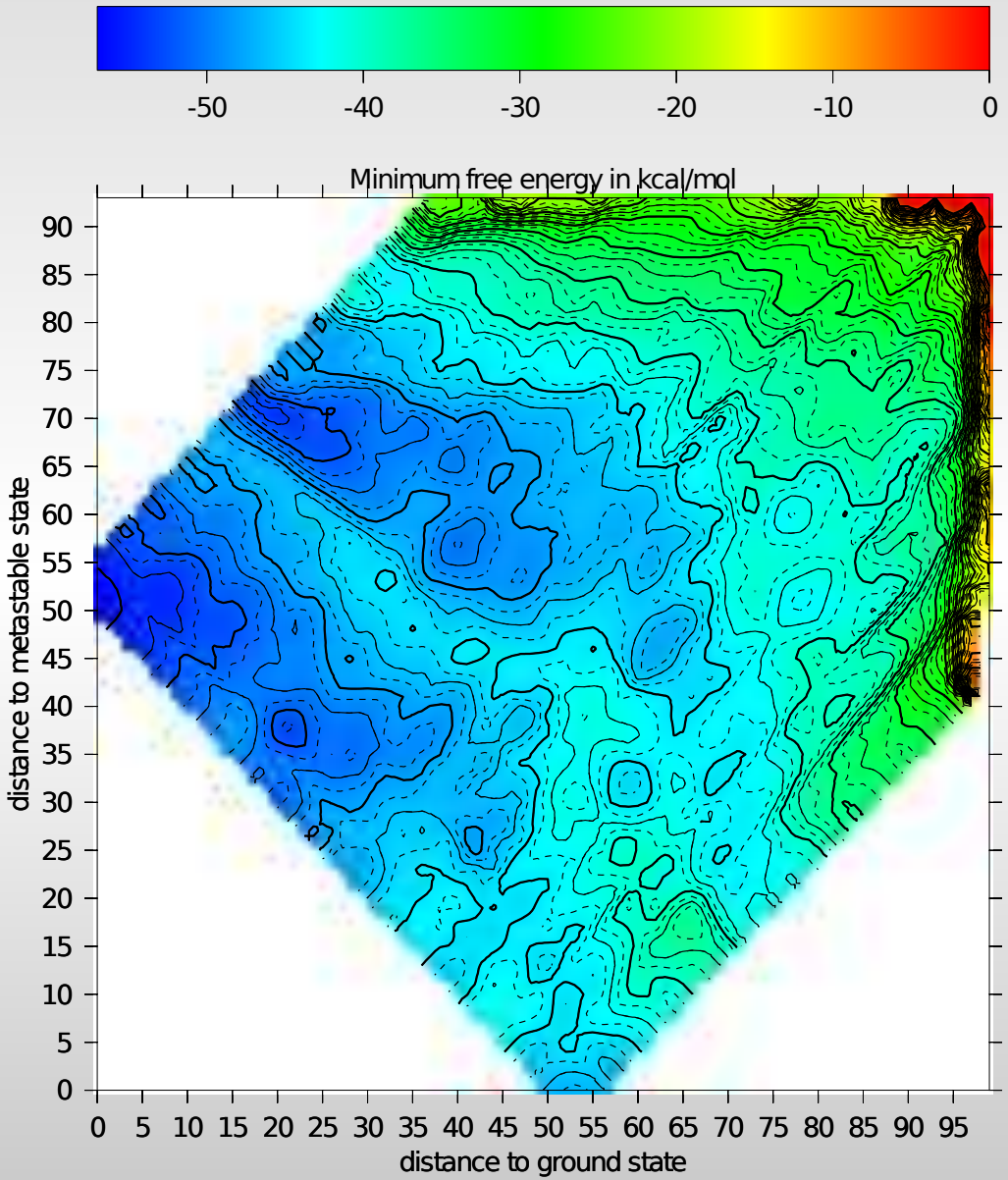
Example 3: 5'-UTR in MS2 - distortion 10^5



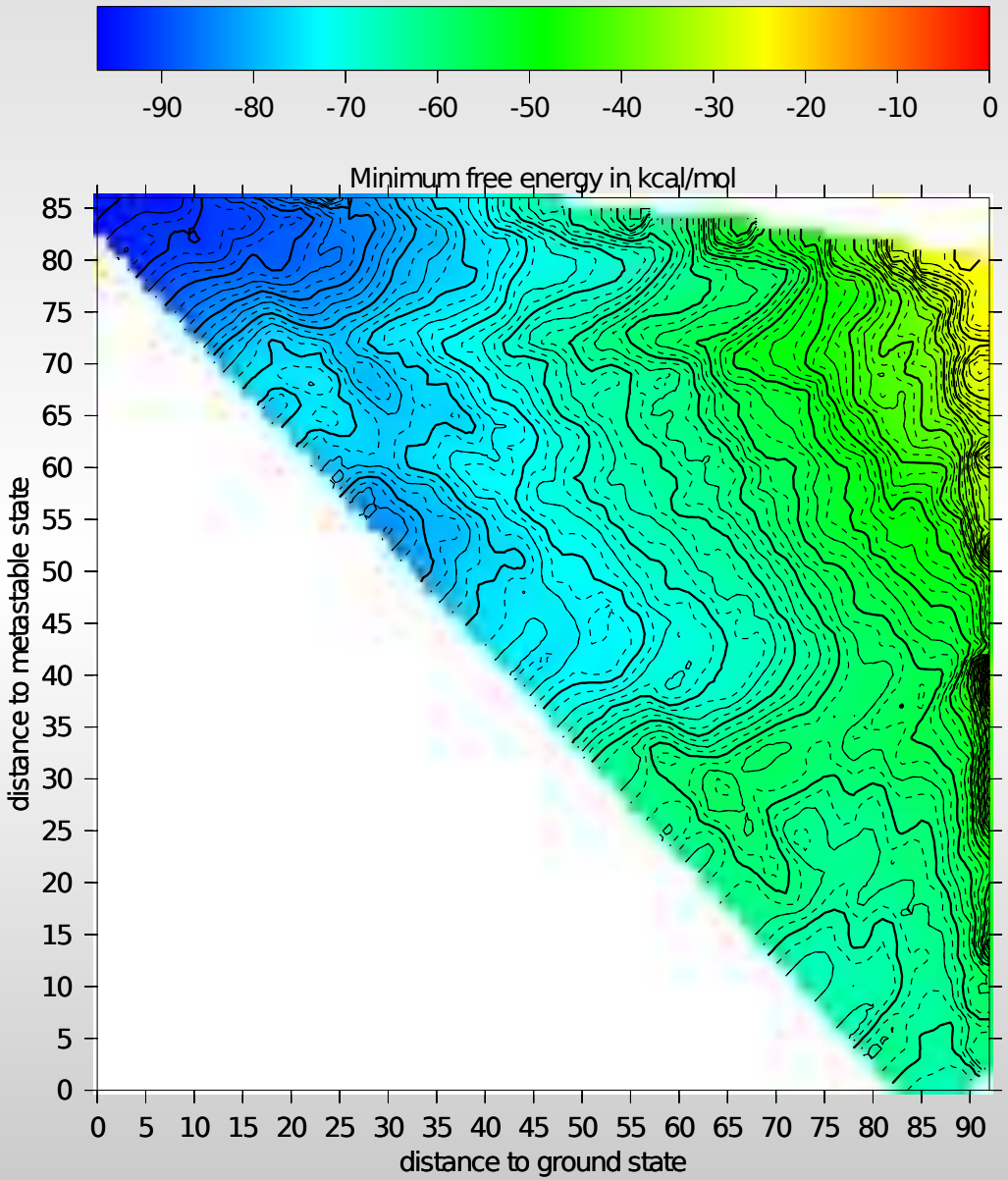
Example 3: 5'-UTR in MS2 - distortion 10^6



Example 3: 5'-UTR in MS2 - RNA2Dfold

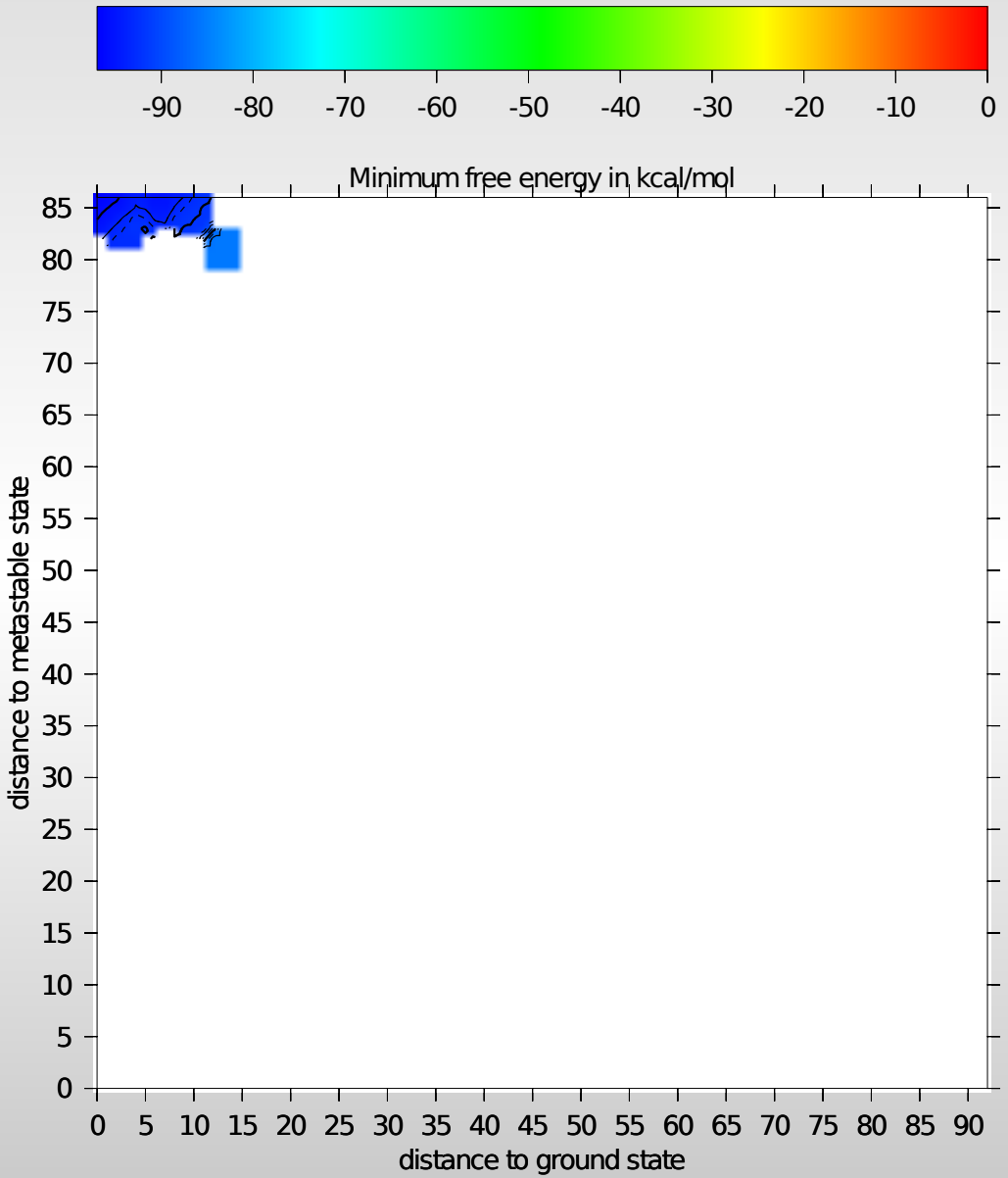


Example 2: SV11 RNA [12] - RNA2Dfold

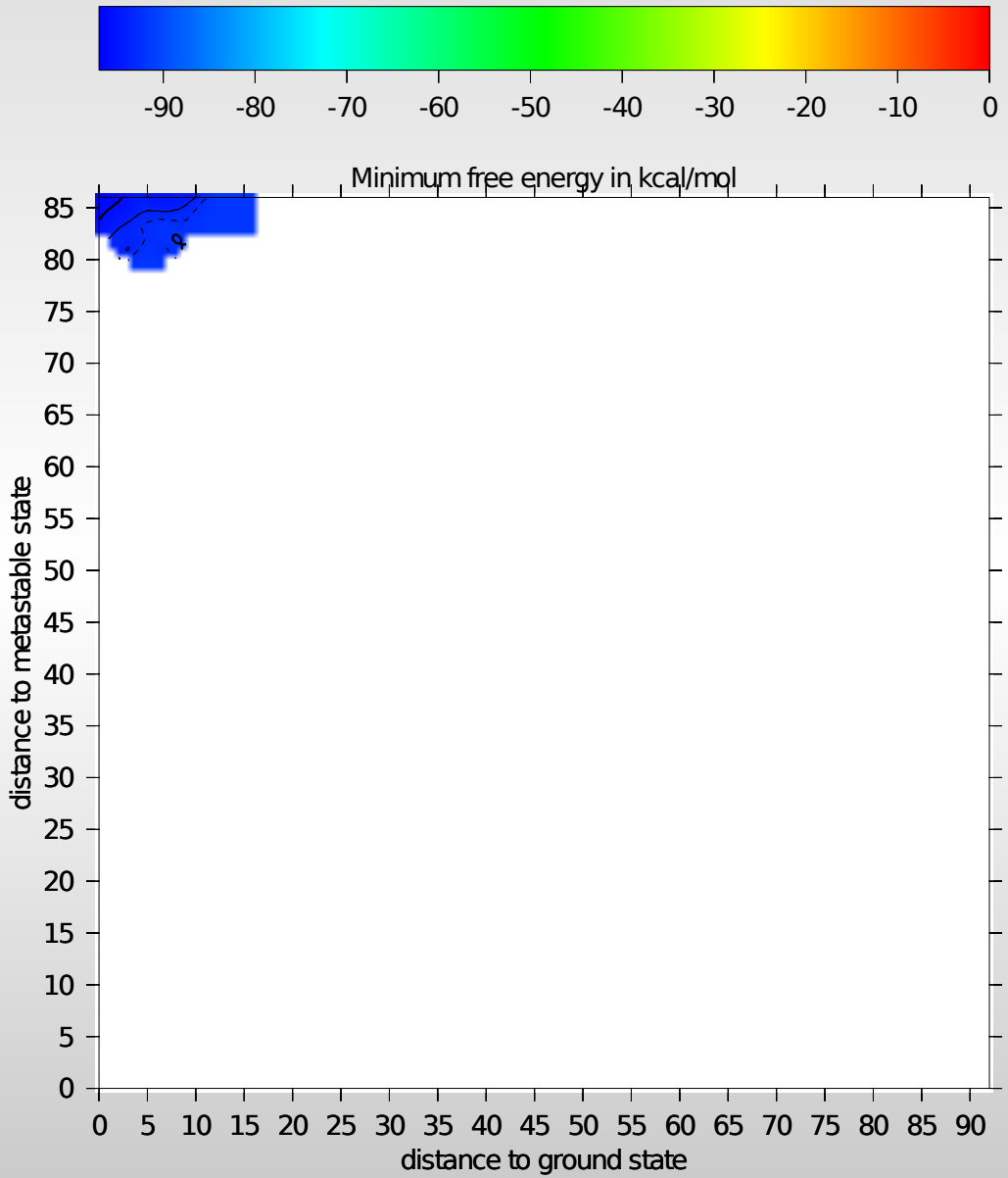


[12] Biebricher et al. 1982, Biebricher and Luce 1992

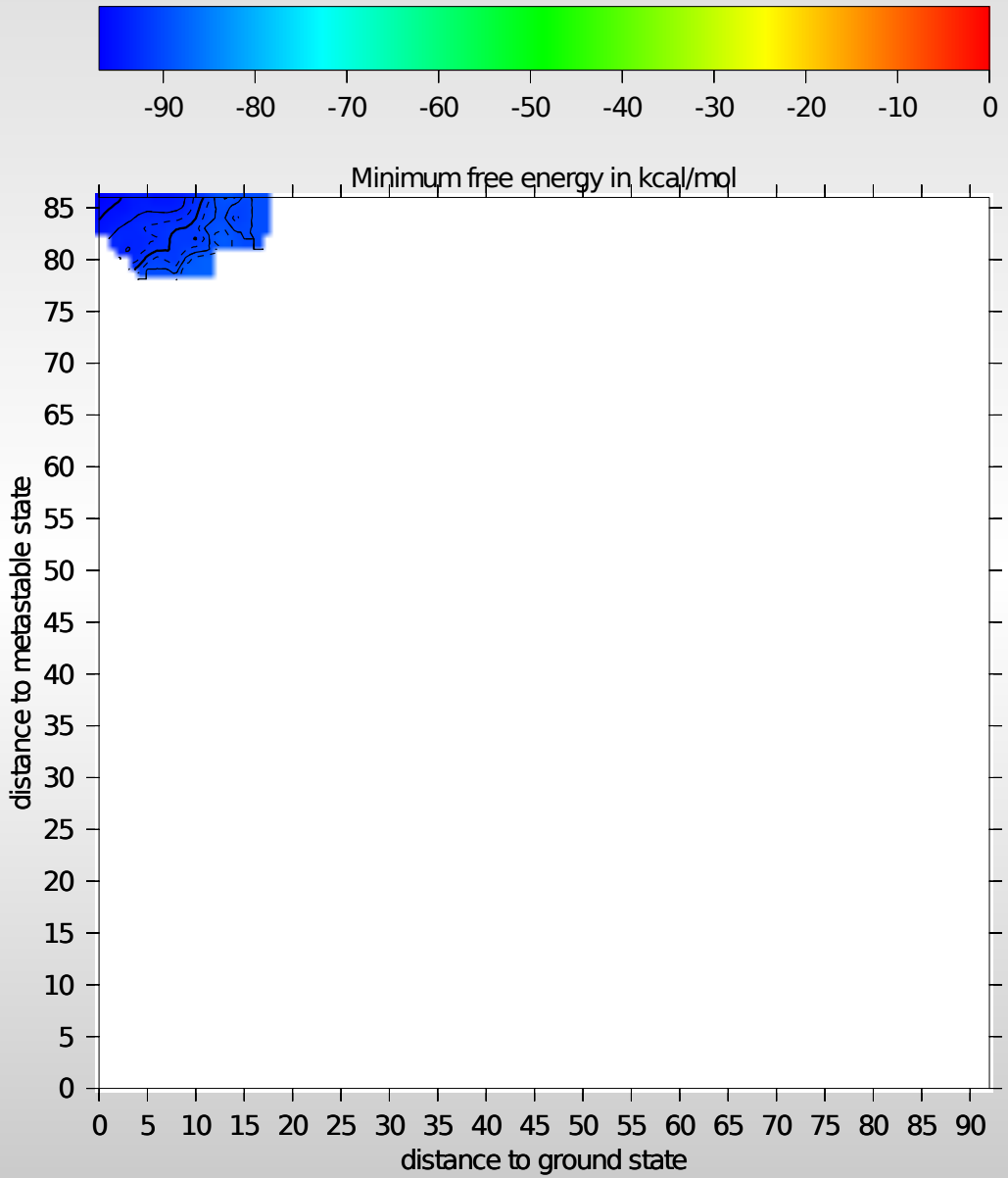
Example 2: SV11 RNA - RNAsubopt -p 10^2



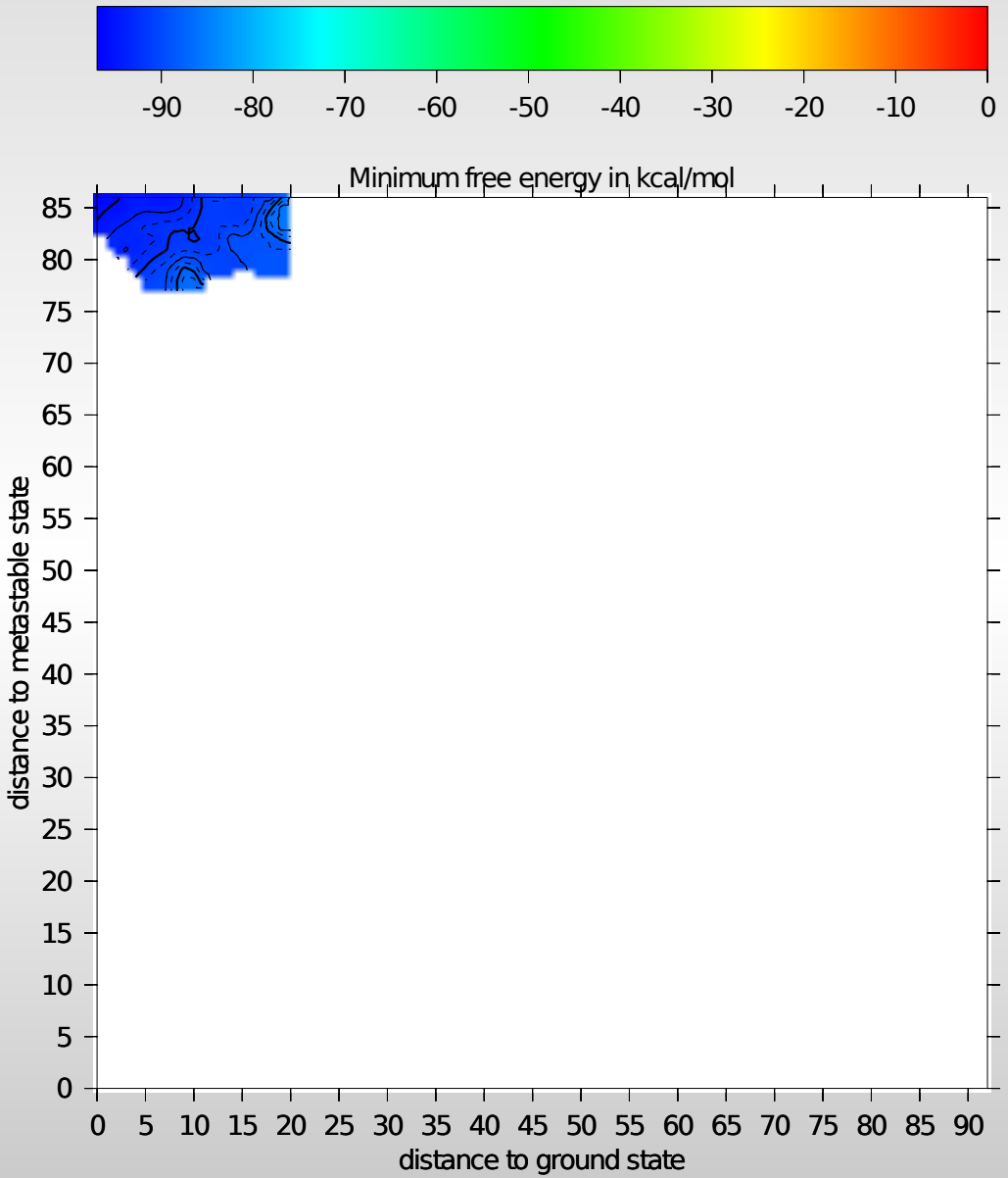
Example 2: SV11 RNA - RNAsubopt -p 10^3



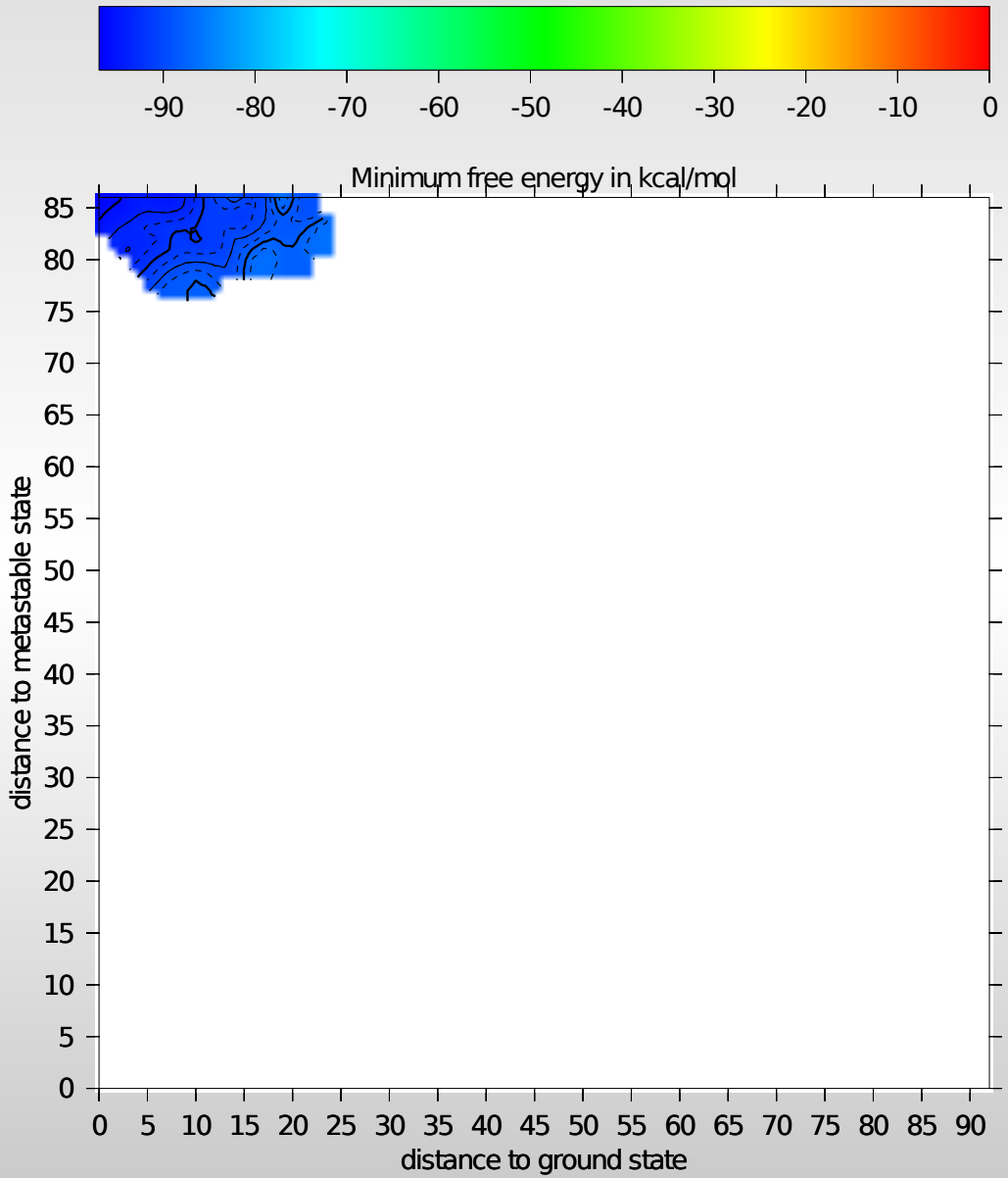
Example 2: SV11 RNA - RNAsubopt -p 10^4



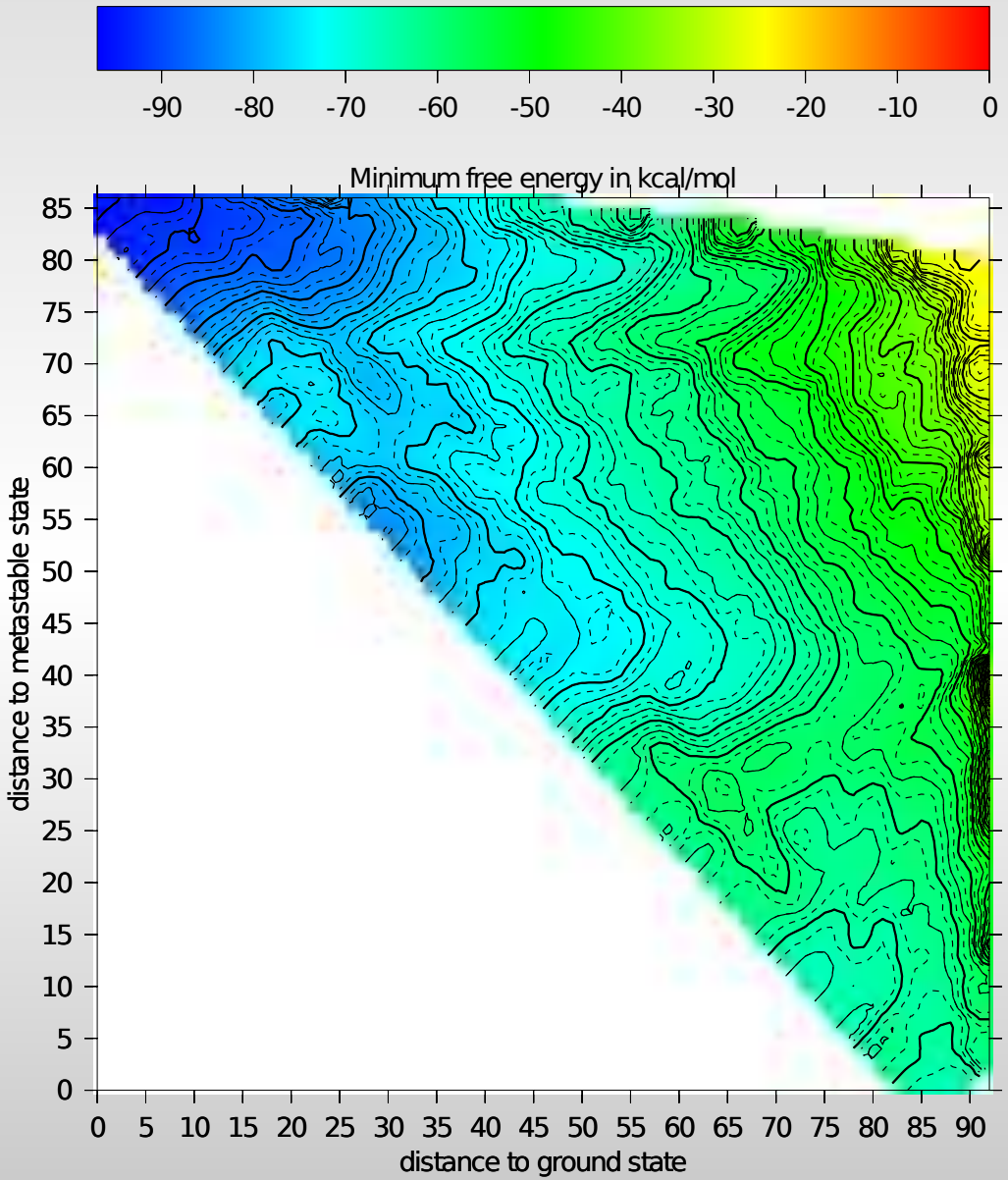
Example 2: SV11 RNA - RNAsubopt -p 10^5



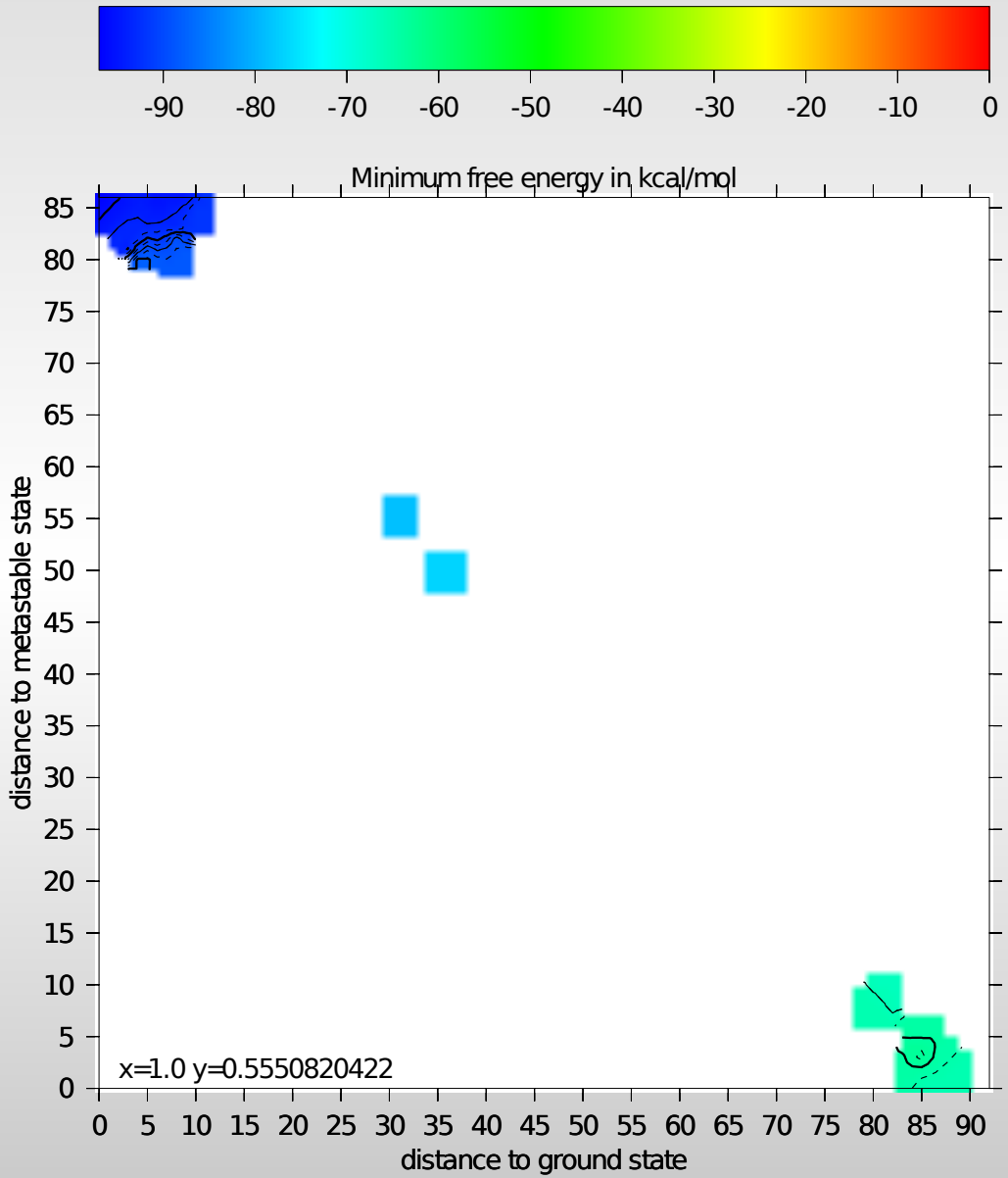
Example 2: SV11 RNA - RNAsubopt -p 10^6



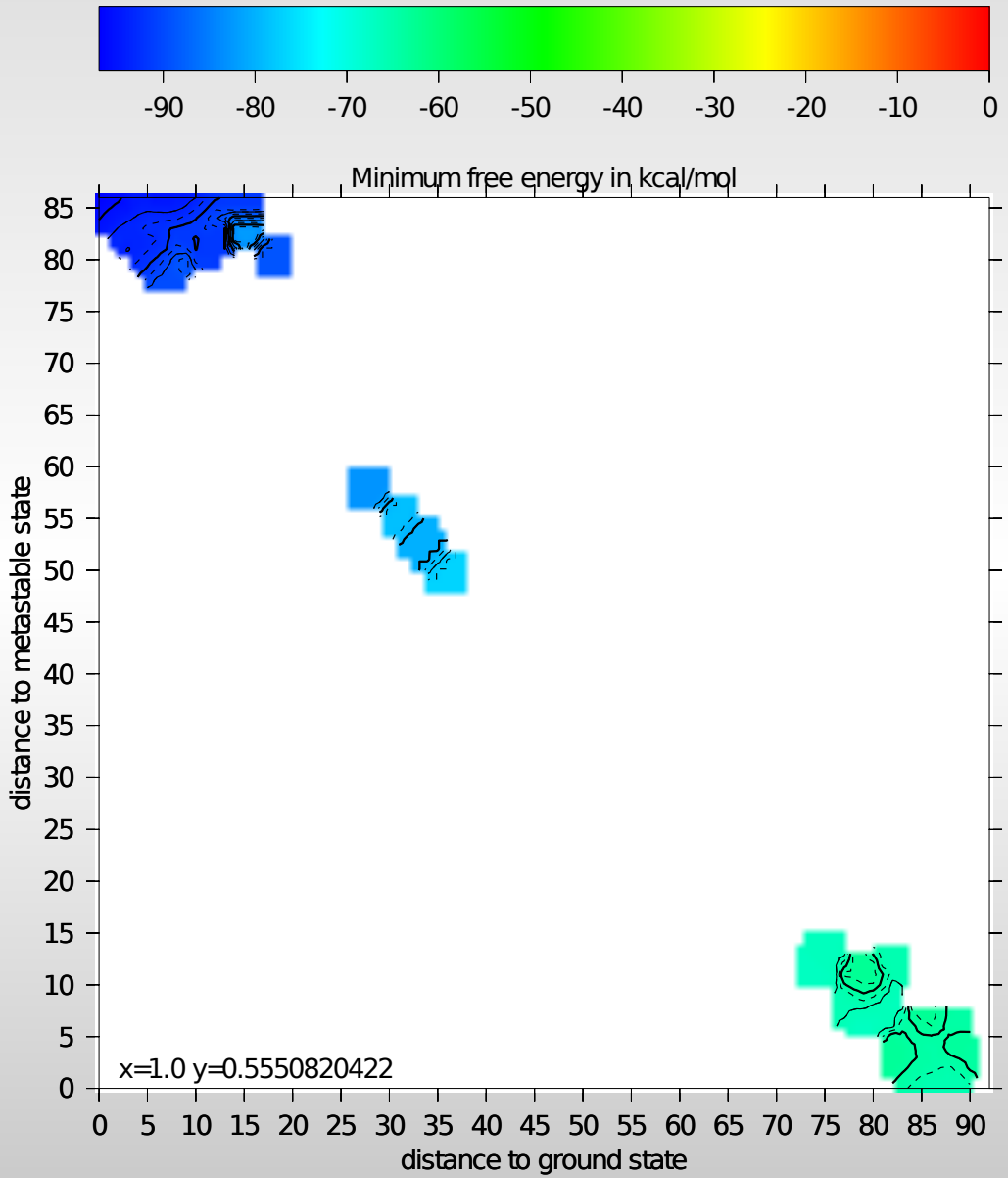
Example 2: SV11 RNA - RNA2Dfold



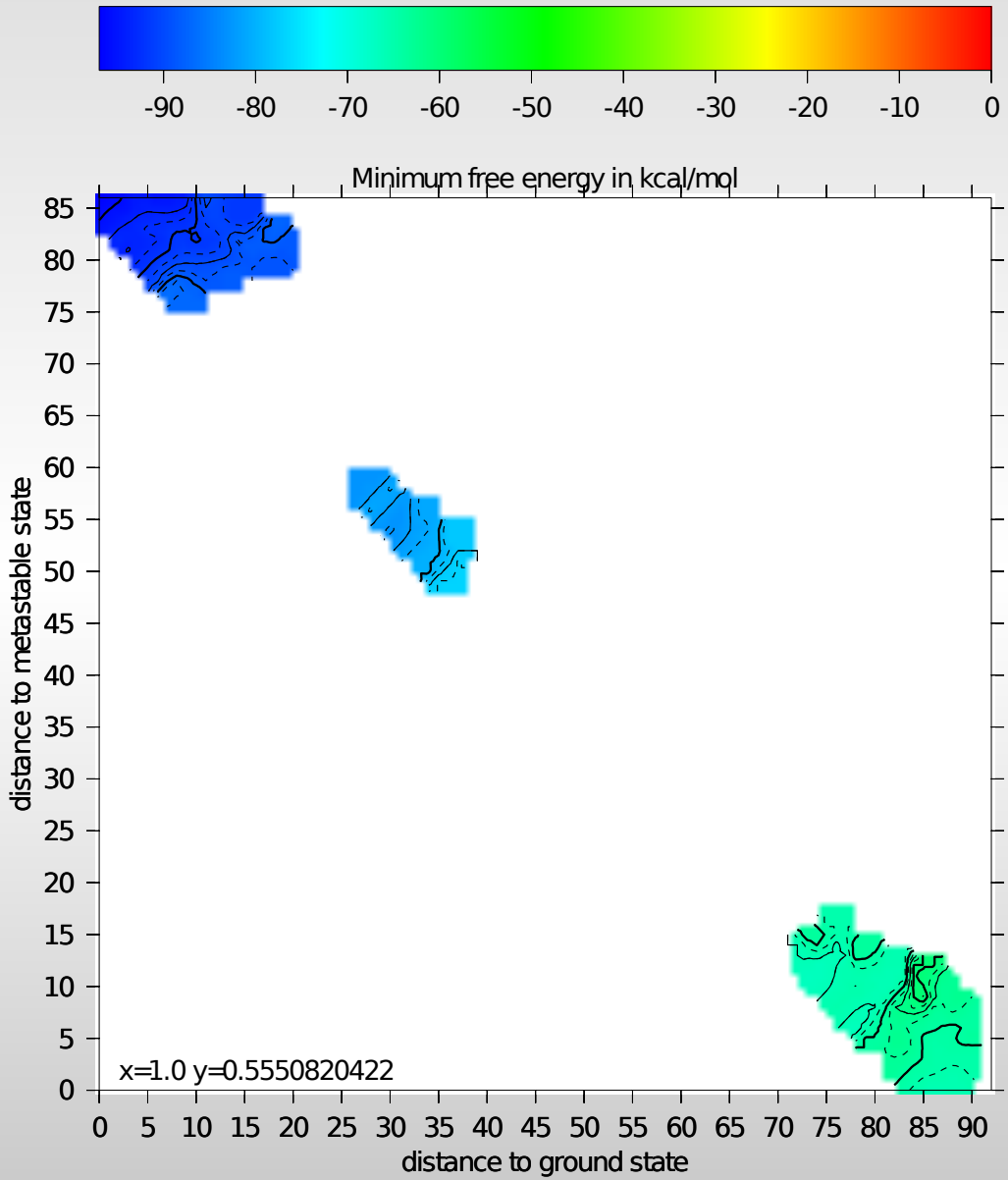
Example 2: SV11 RNA - distortion 10^2



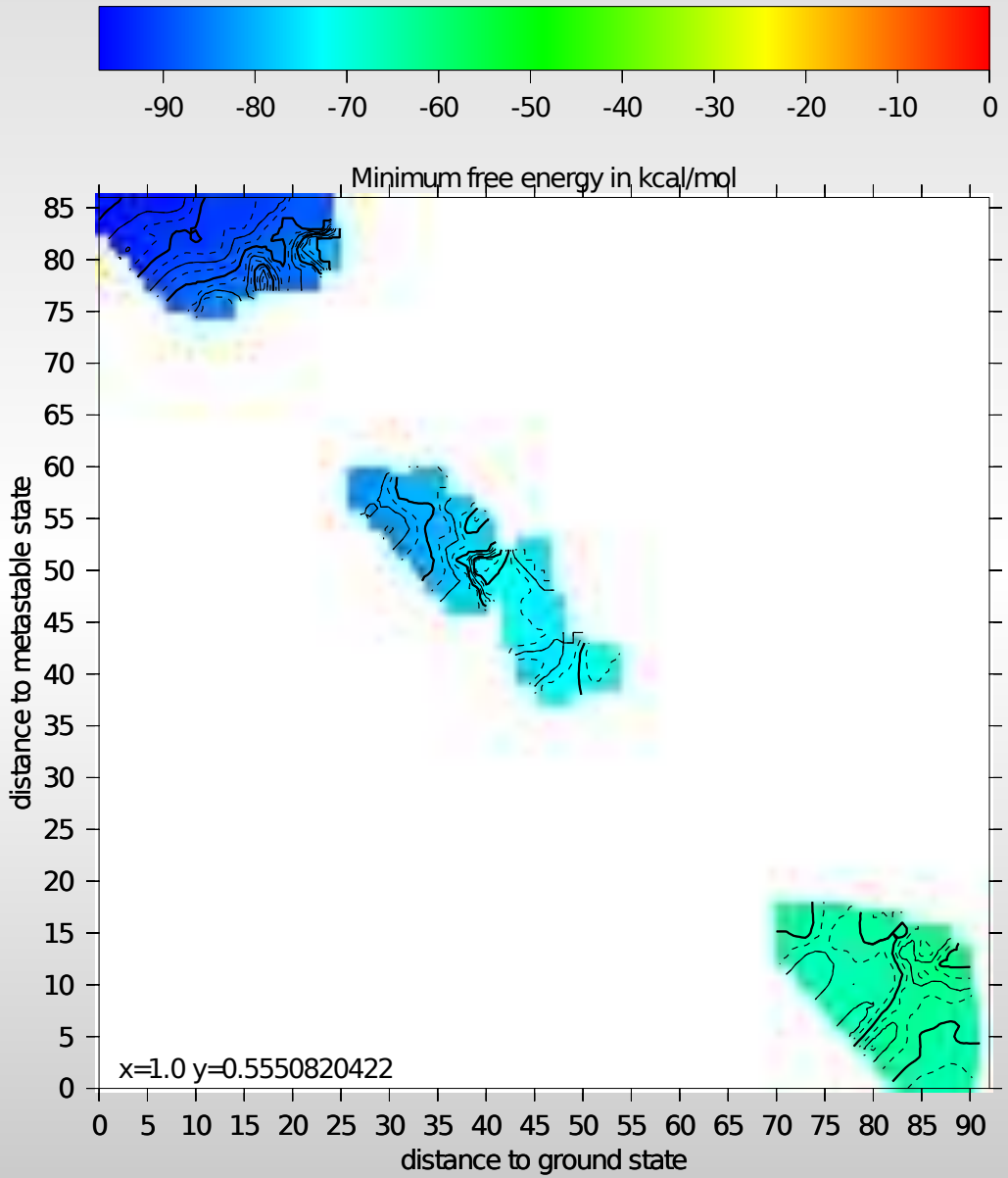
Example 2: SV11 RNA - distortion 10^3



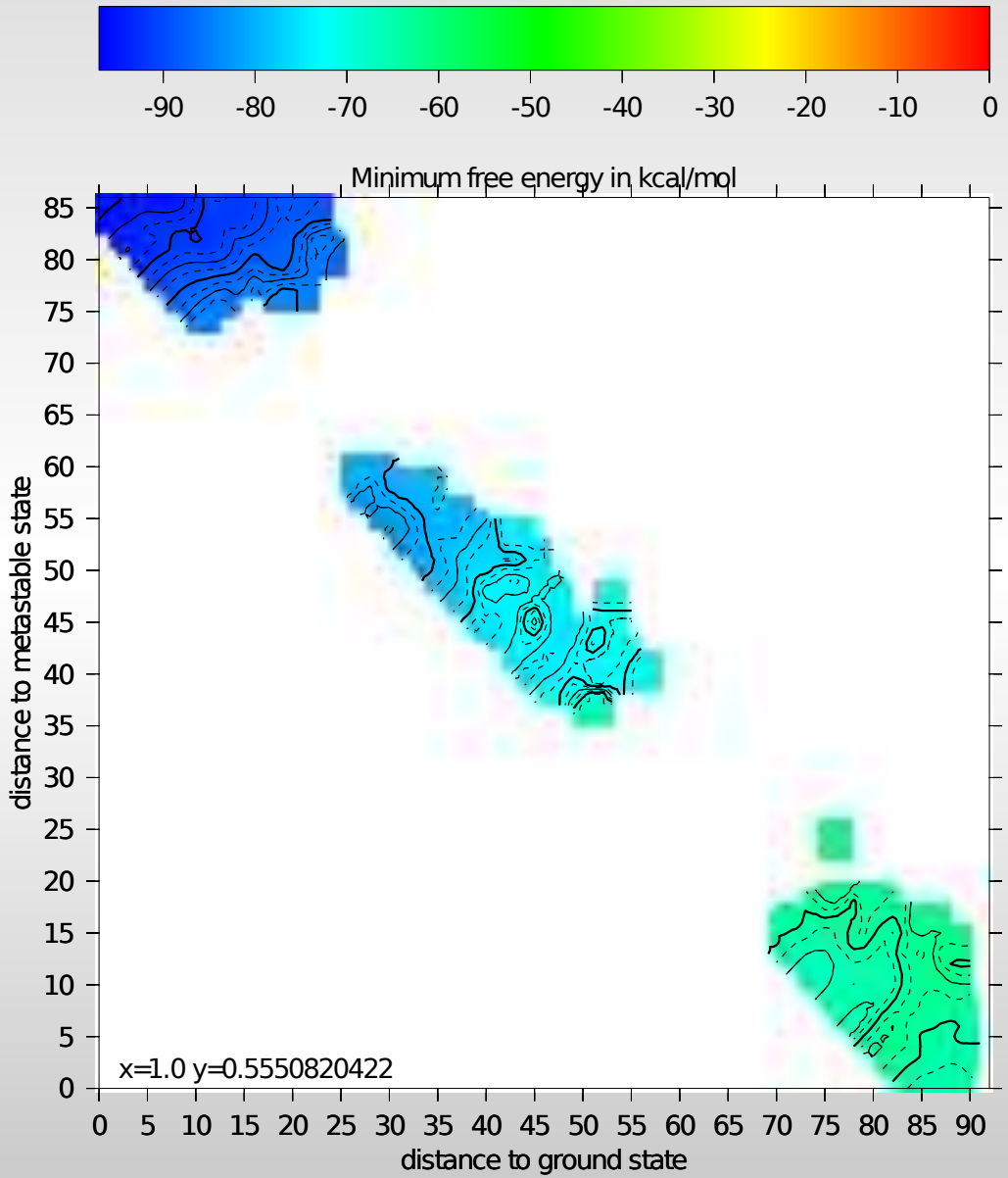
Example 2: SV11 RNA - distortion 10^4



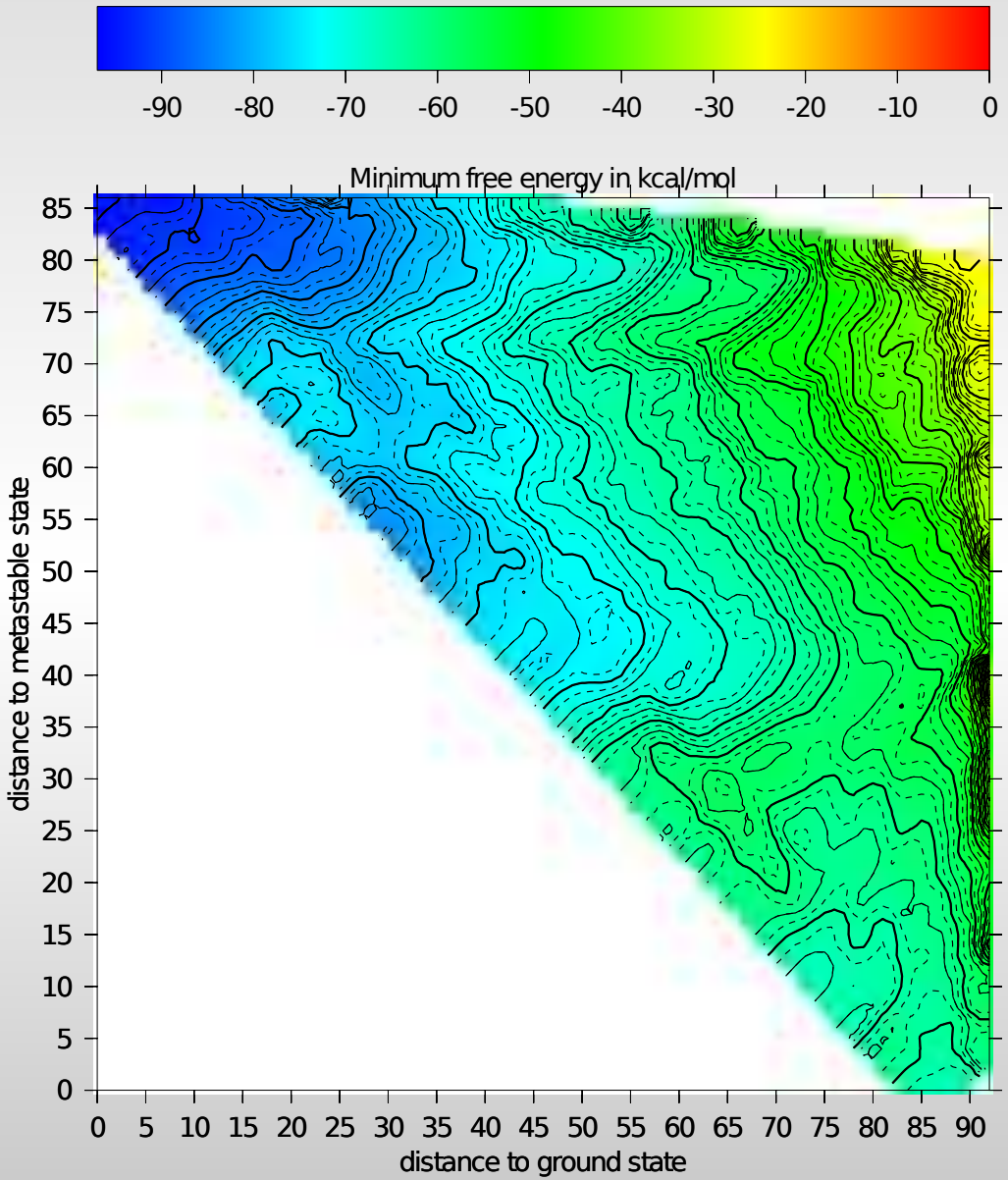
Example 2: SV11 RNA - distortion 10^5



Example 2: SV11 RNA - distortion 10^6



Example 2: SV11 RNA - RNA2Dfold



Under construction

- RNAfold, RNAalifold already support generalized constraints
- ViennaRNA Package v2.2 is scheduled for **this summer**!
- RNAlib API is under change! Backward compatibility until v3
- API will be easy to use:

```
/* obtain a data structure for folding */
vc = vrna_get_fold_compound(sequence, ...);
/* add hard constraints */
vrna_hc_add(vc, constraints_file, ...);
/* add SHAPE reactivity data and apply Mathews conversion
   for pseudo energies */
vrna_sc_add_mathews(vc, shape_data, ...);
/* fold it */
vrna_fold(vc);
```

- Scripting language support (Perl/Python) has fallen behind.

Under construction

- RNAfold, RNAalifold already support generalized constraints
- ViennaRNA Package v2.2 is scheduled for **this summer**!
- RNAlib API is under change! Backward compatibility until v3
- API will be easy to use:

```
/* obtain a data structure for folding */
vc = vrna_get_fold_compound(sequence, ...);
/* add hard constraints */
vrna_hc_add(vc, constraints_file, ...);
/* add SHAPE reactivity data and apply Mathews conversion
   for pseudo energies */
vrna_sc_add_mathews(vc, shape_data, ...);
/* fold it */
vrna_fold(vc);
```

- Scripting language support (Perl/Python) has fallen behind.
  Help with the SWIG interface files would be highly appreciated!

Thanks to

- Dominik Luntzer
- Manja Wachsmuth
- Sven Findeis
- Yann Ponty
- Mario Moerl
- Peter F Stadler
- Ivo L Hofacker

Thank You for your attention!