Combinatorial Optimization in Bioinfo

Lecture 2 – Folding RNA *in silico*

Yann Ponty

AMIBio Team
CNRS & École Polytechnique

. . . or how to make a million bucks by giving change parsimoniously!!

Problem: You have access to unlimited amount of **1**, **20** and **50** cents coins.
A client prefers to travel light, i.e. to **minimize the #coins**.
How to give **N** cents back in change without losing a customer?

Strategy #1: Start with *heaviest* coins, and then complete/fill-up with coins of *decreasing* value.

21 =??

55

60

. . . or how to make a million bucks by giving change parsimoniously!!

Problem: You have access to unlimited amount of **1**, **20** and **50** cents coins.
A client prefers to travel light, i.e. to **minimize the #coins**.
How to give **N** cents back in change without losing a customer?

Strategy #1: Start with *heaviest* coins, and then complete/fill-up with coins of *decreasing* value.

21 =  + 

55??

60

. . . or how to make a million bucks by giving change parsimoniously!!

Problem: You have access to unlimited amount of **1**, **20** and **50** cents coins.
A client prefers to travel light, i.e. to **minimize the #coins**.
How to give **N** cents back in change without losing a customer?

Strategy #1: Start with *heaviest* coins, and then complete/fill-up with coins of *decreasing* value.

21 = 

55 = 

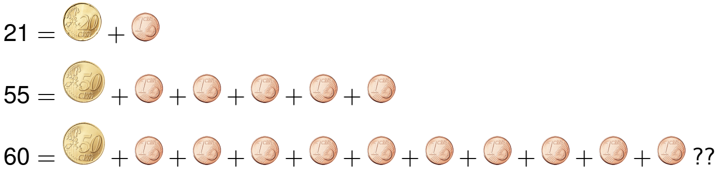
60??

Foreword . . .

. . . or how to make a million bucks by giving change parsimoniously!!

Problem: You have access to unlimited amount of **1**, **20** and **50** cents coins.
A client prefers to travel light, i.e. to **minimize the #coins**.
How to give **N** cents back in change without losing a customer?

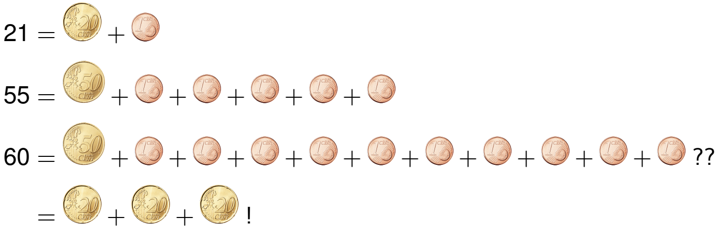
Strategy #1: Start with *heaviest* coins, and then complete/fill-up with coins of *decreasing* value.



. . . or how to make a million bucks by giving change parsimoniously!!

Problem: You have access to unlimited amount of **1**, **20** and **50** cents coins.
A client prefers to travel light, i.e. to **minimize the #coins**.
How to give **N** cents back in change without losing a customer?

Strategy #1: Start with *heaviest* coins, and then complete/fill-up with coins of *decreasing* value.



Problem *a priori (?!)* non-solvable using such a *greedy* approach, as a (simpler) problem is already NP-complete (thus Efficient solution ⇒ 1M$).

Strategy #2: Brute force enumeration $\rightarrow$ #Coins^N (Ouch!)

Strategy #3: The following recurrence gives the minimal number of coins:

$Min\#Coins(N) = \text{Min}$ of:

-  $\rightarrow 1 + Min\#Coins(N-1)$
-  $\rightarrow 1 + Min\#Coins(N-20)$
-  $\rightarrow 1 + Min\#Coins(N-50)$

With some memory (N intermediate computations), the minimum number of coins can be obtained after $N\times$#Coins operations. An actual set of coins can be reconstructing by **tracing back** the choices performed at each stage, leading to the minimum.

Remark: We still haven't won the million, as N has **exponential value compared to the length of its encoding**, so the algorithm does not qualify as *efficient* (i.e. polynomial).

Still, this approach is much more efficient than a brute-force enumeration:

$\Rightarrow$ Dynamic programming.

Strategy #2: Brute force enumeration $\rightarrow$ #Coins^N (Ouch!)

Strategy #3: The following recurrence gives the minimal number of coins:

$Min\#Coins(N) = \text{Min}$

-  $\rightarrow 1 + Min\#Coins(N-1)$
-  $\rightarrow 1 + Min\#Coins(N-20)$
-  $\rightarrow 1 + Min\#Coins(N-50)$

With some memory (N intermediate computations), the minimum number of coins can be obtained after $N\times$#Coins operations. An actual set of coins can be reconstructing by **tracing back** the choices performed at each stage, leading to the minimum.

Remark: We still haven't won the million, as N has **exponential value compared to the length of its encoding**, so the algorithm does not qualify as *efficient* (i.e. polynomial).

Still, this approach is much more efficient than a brute-force enumeration:

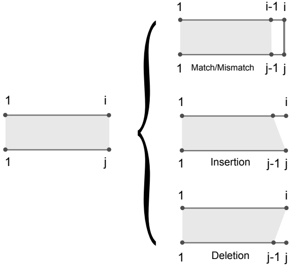
$\Rightarrow$ Dynamic programming.

Dynamic programming = General optimization technique.
Prerequisite: Optimal solution for problem P can be derived from solutions to strict sub-problems of P.

Bioinformatics :

- Discrete solution space (alignments, structures. . .)
- **+** Additively-inherited objective function (cost, log-odd score, energy. . .)
- $\Rightarrow$ Efficient dynamic programming scheme

Example: Local Alignment(Smith/Waterman)



$$\begin{aligned} W(i,0) &= 0 \\ W(0,j) &= 0 \\ W(i,j) &= \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases} \end{aligned}$$

Dynamic programming scheme defines a space of (sub)problems and a recurrence that relates the score of a problem to that of smaller problems.

Given a scheme, two steps :

- **Matrix filling:** Computation and tabulation of best scores (Computed from smaller problems to larger ones).
- **Traceback:** Reconstruct best solution from contributing subproblems.

Complexity of algorithm depends on:

- **Cardinality** of sub-problem space
- **Number of alternatives** considers at each step (#Terms in recurrence)

Smith&Waterman example:

- i: $1 \to n+1 \Rightarrow \Theta(n)$
- j: $1 \to m+1 \Rightarrow \Theta(m)$
- 3 operations at each step

$\Rightarrow \Theta(m.n)$ time/memory

$$
\begin{aligned}
W(i,0) &= 0\\
W(0,j) &= 0\\
W(i,j) &= \max \begin{cases} W(i-1,j-1)+m_{i,j}\\ W(i-1,j)+p_i\\ W(i,j-1)+p_d \end{cases}
\end{aligned}
$$

Necessary properties:

- **Correctness:** $\forall$ sub-problem, the computed value must indeed maximize the objective function .

Proofs usually inductive, and quite technical, but very systematic.

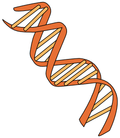
Desirable properties of DP schemes:

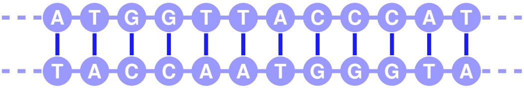
- **Completeness** of space of solutions generated by decomposition. Algorithmic tricks, by *cutting branches*, may violate this property.
- **Unambiguity:** Each solution is generated at most once.

$\Rightarrow$ Under these properties, one can **enumerate** solution space.

DNA

$\{A, C, G, T\}^\star$

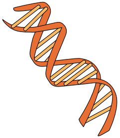


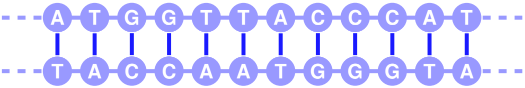


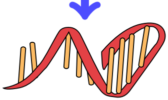
Fundamental *dogma* of molecular biology

DNA

$\{A, C, G, T\}^\star$





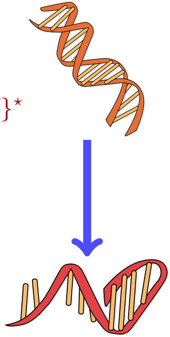


RNAs

$\{A, C, G, U\}^\star$

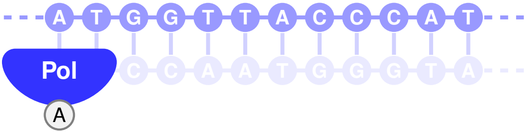
DNA

$\{A, C, G, T\}^{\star}$



RNAs

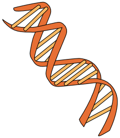
$\{A, C, G, U\}^{\star}$

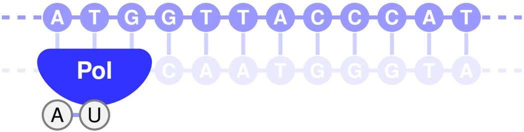


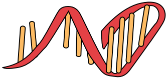
Fundamental *dogma* of molecular biology

DNA

$\{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}^\star$





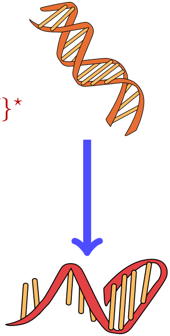


RNAs

$\{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{U}\}^\star$

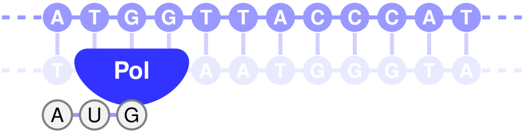
DNA

$\{A, C, G, T\}^\star$



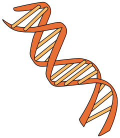
RNAs

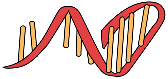
$\{A, C, G, U\}^\star$



DNA

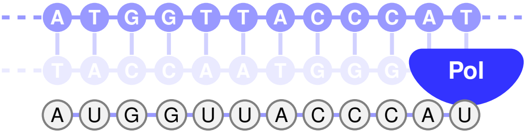
$\{A, C, G, T\}^{\star}$





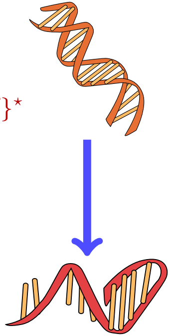
RNAs

$\{A, C, G, U\}^{\star}$



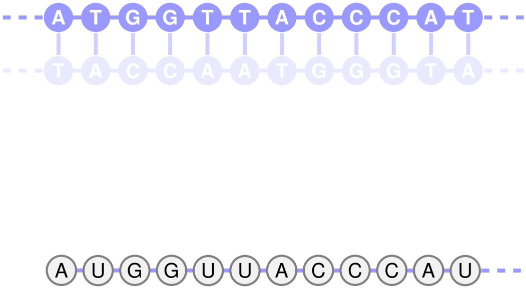
DNA

$\{A, C, G, T\}^\star$



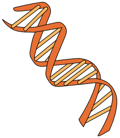
RNAs

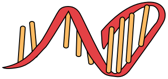
$\{A, C, G, U\}^\star$



DNA

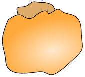
$\{A, C, G, T\}^\star$





RNAs

$\{A, C, G, U\}^\star$

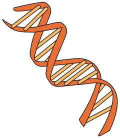


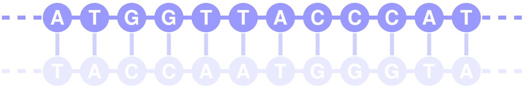
Proteins

$\underbrace{\{\text{Ala}, \text{Arg}, \ldots, \text{Val}\}^\star}_{20^+ \text{ Amino acids}}$

DNA

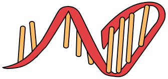
$\{A, C, G, T\}^\star$

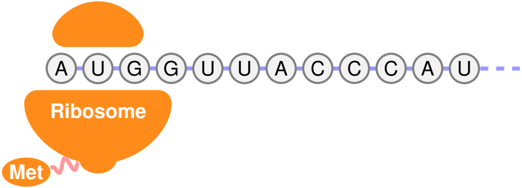




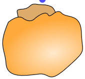
RNAs

$\{A, C, G, U\}^\star$



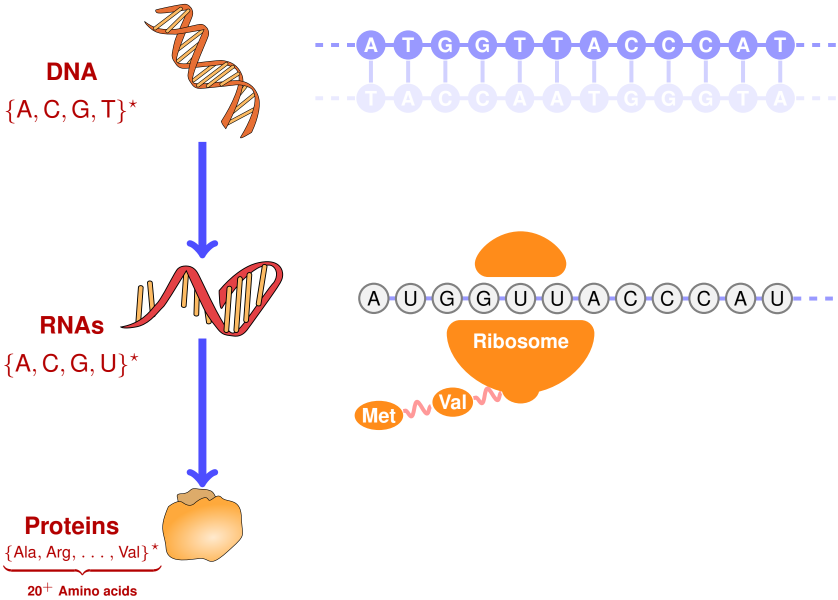


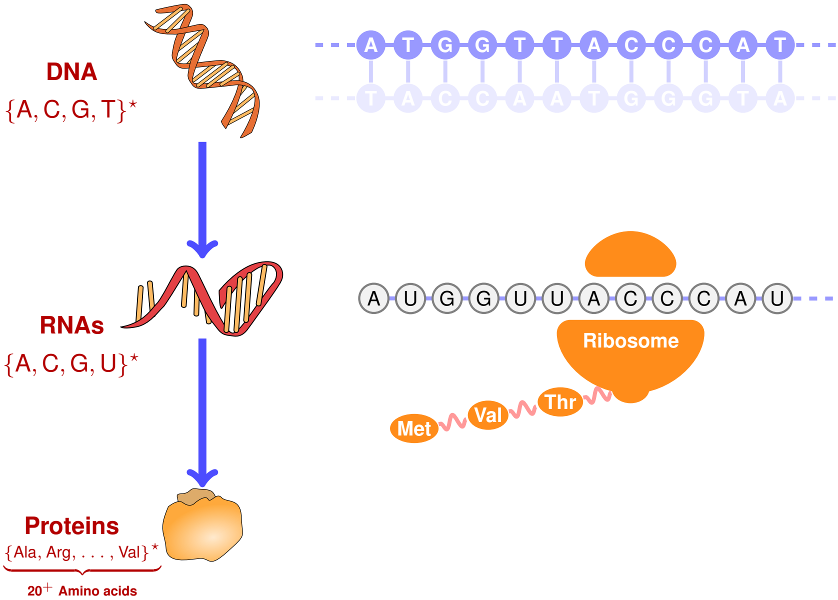
Proteins



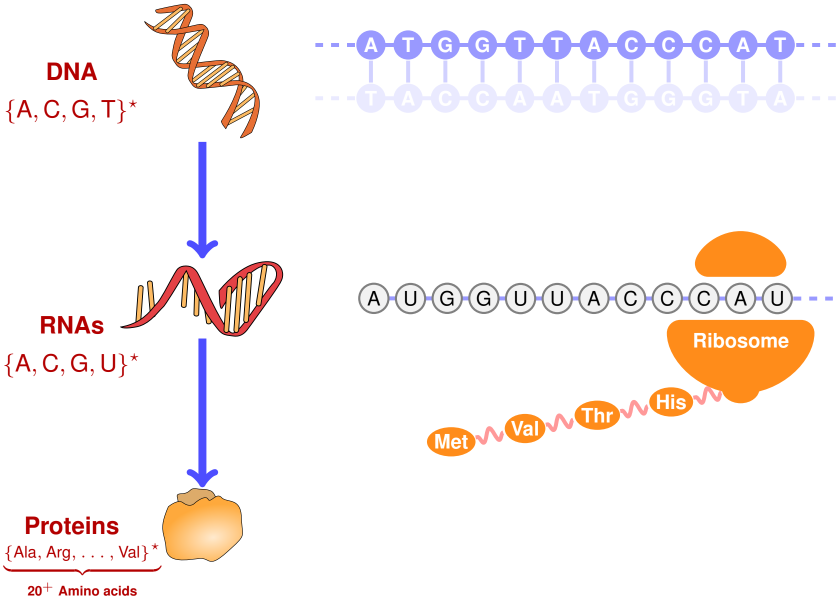
$\underbrace{\{\text{Ala, Arg, . . . , Val}\}^\star}$

20^+ Amino acids

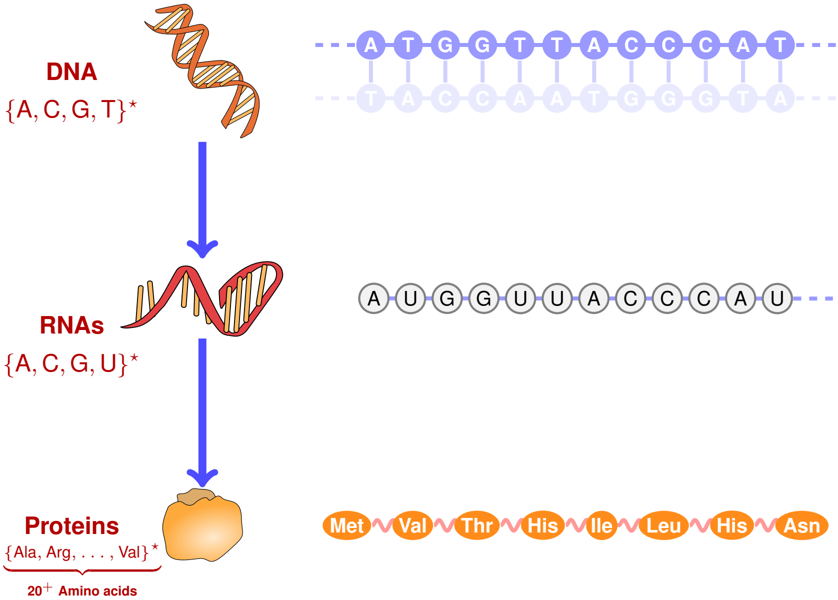

DNA
{A, C, G, T}*
A T G G T T A C C C A T
T A C C A A T G G G T A
RNAs
{A, C, G, U}*
A U G G U U A C C C A U
Ribosome
Met
Val
Proteins
{Ala, Arg, . . . , Val}*
20+ Amino acids

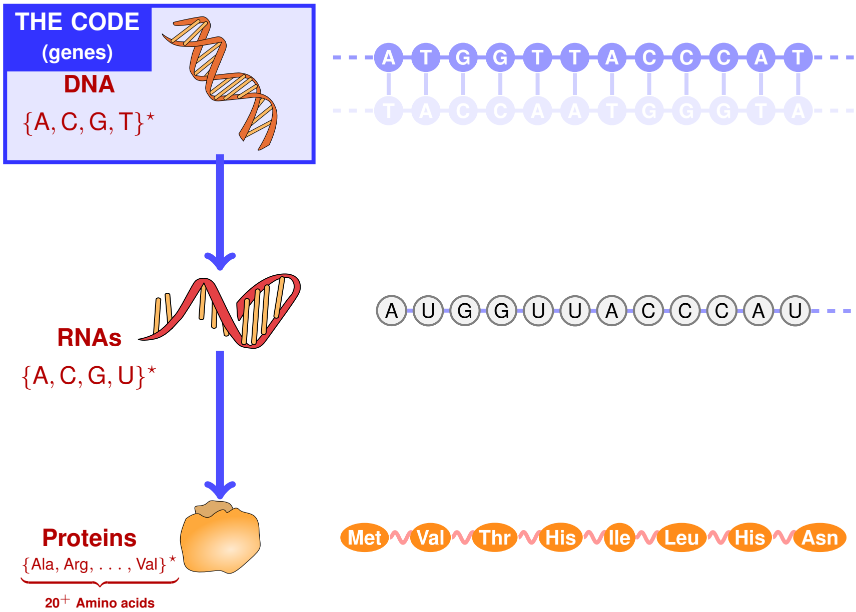

DNA
{A, C, G, T}*
A T G G T T A C C C A T
T A C C A A T G G G T A
RNAs
{A, C, G, U}*
A U G G U U A C C C A U
Ribosome
Met
Val
Thr
Proteins
{Ala, Arg, . . . , Val}*
20+ Amino acids

Fundamental *dogma* of molecular biology

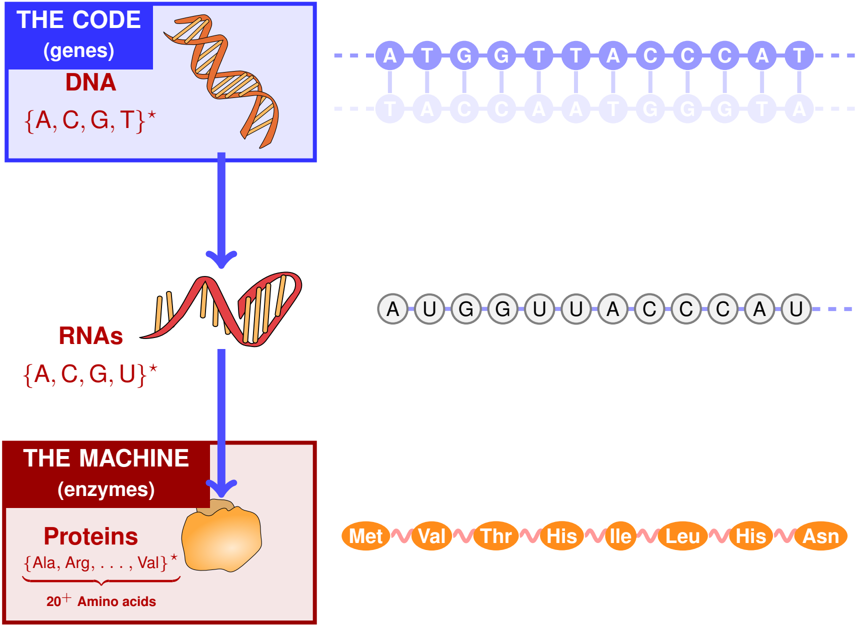


Fundamental *dogma* of molecular biology

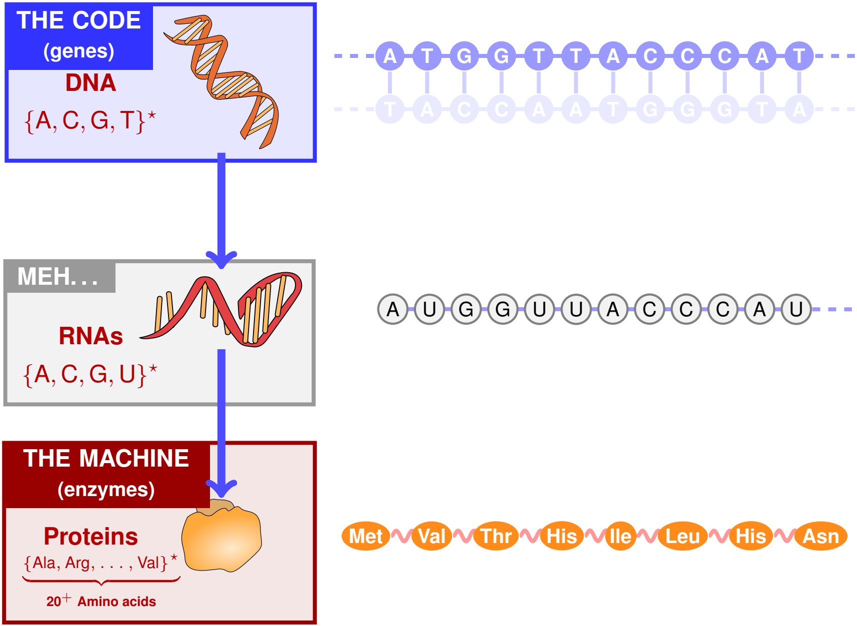


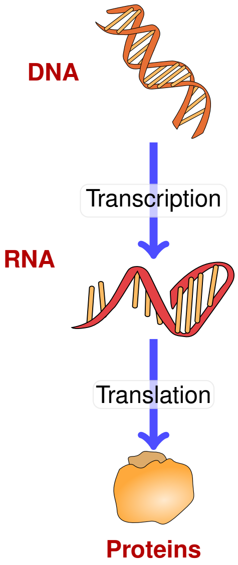

THE CODE
(genes)
DNA
{A, C, G, T}*
A T G G T T A C C C A T
T A C C A A T G G G T A
RNAs
{A, C, G, U}*
A U G G U U A C C C A U
Proteins
{Ala, Arg, . . . , Val}*
20+ Amino acids
Met Val Thr His Ile Leu His Asn

Fundamental *dogma* of molecular biology

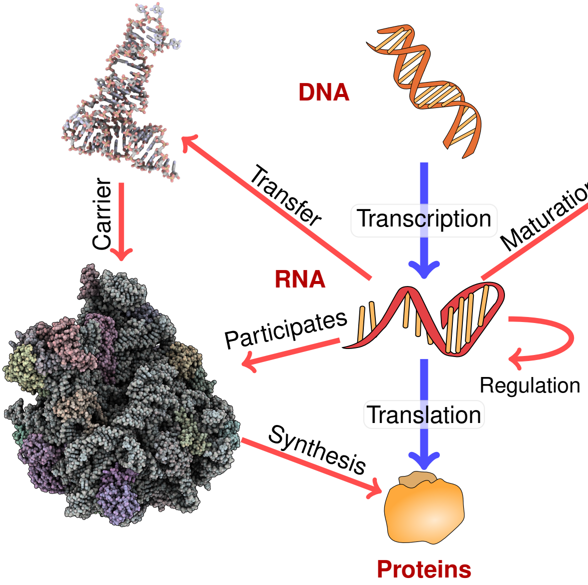


Fundamental *dogma* of molecular biology



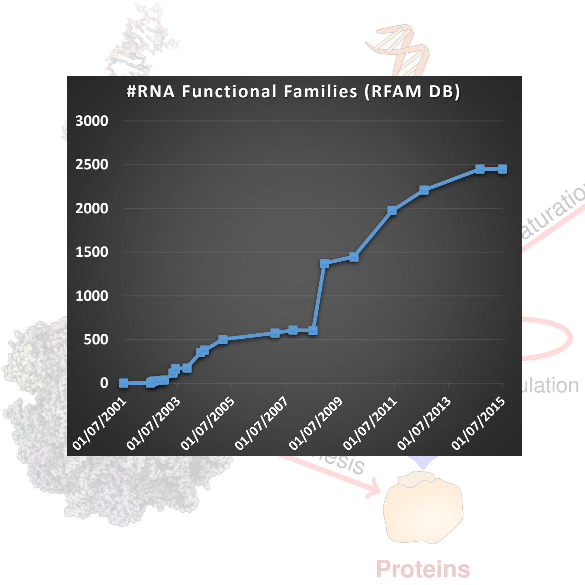

DNA
Transcription
RNA
Translation
Proteins

Fundamental *dogma* of molecular biology



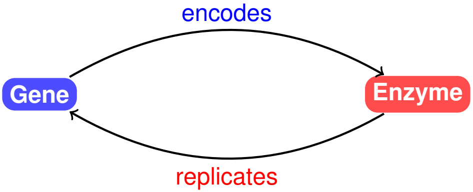
RNA functions

- Messenger
- Translation
- Regulation
- Enzyme
- Catalytic
- ...



RNA functions

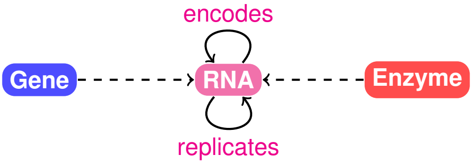
- Messenger
- Translation
- Regulation
- Enzyme
- Catalytic
- . . .



A gene big enough to specify an enzyme would be too big to replicate accurately without the aid of an enzyme of the very kind that it is trying to specify. So the system *apparently cannot get started*.

[. . .] This is the RNA World. To see how plausible it is, we need to look at why proteins are good at being enzymes but bad at being replicators; at why DNA is good at replicating but bad at being an enzyme; and finally why *RNA might just be good enough at both roles to break out of the Catch-22.*

R. Dawkins. *The Ancestor's Tale: A Pilgrimage to the Dawn of Evolution*



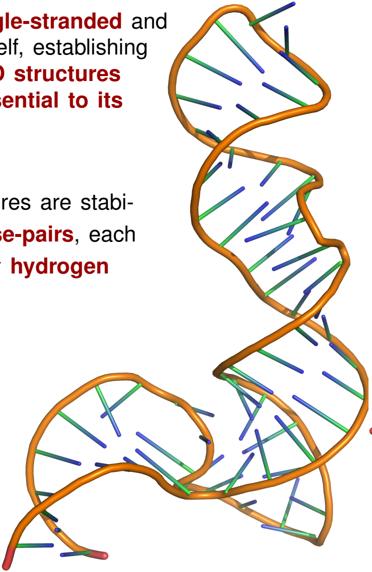
A gene big enough to specify an enzyme would be too big to replicate accurately without the aid of an enzyme of the very kind that it is trying to specify. So the system *apparently cannot get started*.

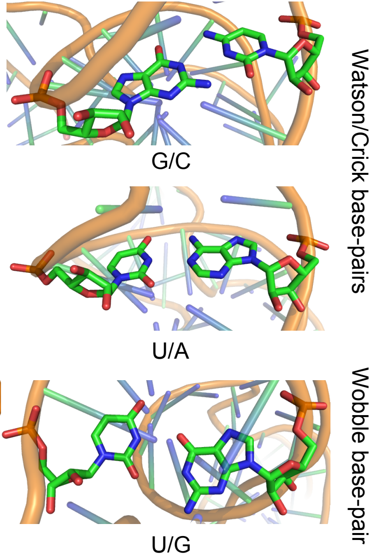
[...] This is the RNA World. To see how plausible it is, we need to look at why proteins are good at being enzymes but bad at being replicators; at why DNA is good at replicating but bad at being an enzyme; and finally why *RNA might just be good enough at both roles to break out of the Catch-22.*

R. Dawkins. *The Ancestor's Tale: A Pilgrimage to the Dawn of Evolution*

RNA is **single-stranded** and **folds** on itself, establishing **complex 3D structures** that are **essential to its function(s)**.

RNA structures are stabilized by **base-pairs**, each mediated by **hydrogen bonds**.



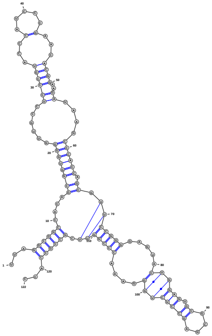


Canonical base-pairs

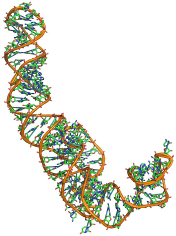
Three[1] levels of representation:

UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA
CC

Primary structure



Secondary structure



Tertiary structure

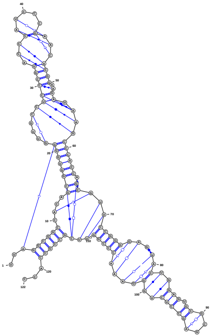
Source: 5s rRNA (PDB 1K73:B)

[1] Well, mostly...

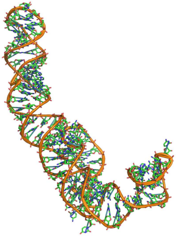
Three[1] levels of representation:

UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA
CC

Primary structure



Secondary$^+$ structure

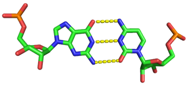


Tertiary structure

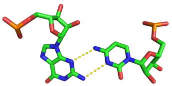
Source: 5s rRNA (PDB 1K73:B)

[1] Well, mostly...

- **Non-canonical base-pairs**
  Any base-pair other than {(A-U), (C-G), (G-U)}
  Or interacting on non-standard edge (≠ WC/WC-Cis) [LW01].

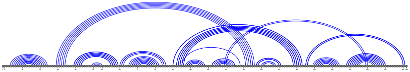


Canonique CG pair(WC/WC-Cis)



Non-canonique CG pair (Sugar/WC-Trans)

- **Pseudoknots (PKs)**



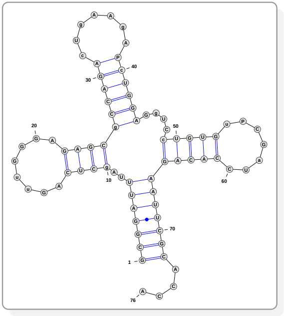
Pseudoknoted structure of group I ribozyme (PDBID: 1Y0Q:A)

Considering PKs may lead to better predictions, **but**:

- Some PK conformations are simply unfeasible;
- Folding *in silico* with general pseudoknots is NP-complete [LP00];

Still, folding on restricted classes of conformations seems promising [CDR+04].

Various representations for a versatile biomolecule



Outer-planar graphs

Hamiltonian-path, $\Delta(G) \leq 3$, 2-connected*

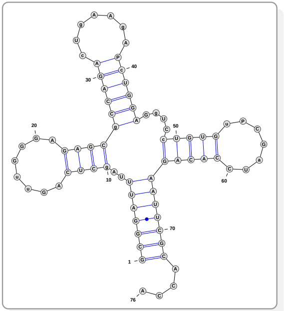
Supporting intuitions

Different representations

Common combinatorial structure

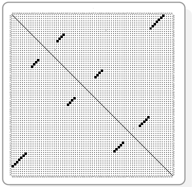
*Additional steric constraints

Various representations for a versatile biomolecule



Outer-planar graphs

Hamiltonian-path, $\Delta(G) \leq 3$, 2-connected*



Dot plots

Adjacency matrices*

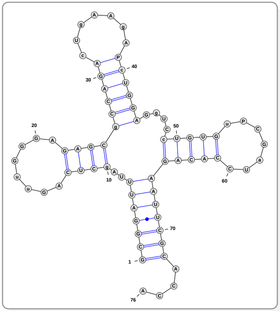
Supporting intuitions

Different representations

Common combinatorial structure

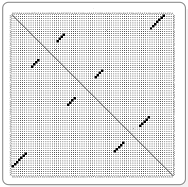
*Additional steric constraints

Various representations for a versatile biomolecule



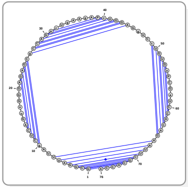
Outer-planar graphs

Hamiltonian-path, $\Delta(G) \leq 3$, 2-connected*



Dot plots

Adjacency matrices*



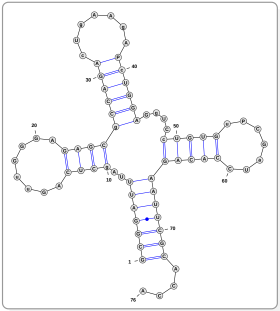
Non-crossing arc diagrams*

Supporting intuitions

Different representations

Common combinatorial structure

*Additional steric constraints

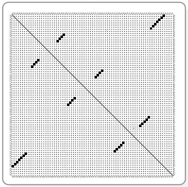


Outer-planar graphs

Hamiltonian-path, $\Delta(G) \leq 3$, 2-connected*

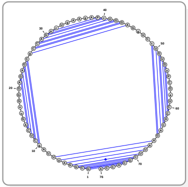
```
(((((((..((((........))))((((((.......))))))....(((((.......))))))))))))....
```

Motzkin words*



Dot plots

Adjacency matrices*



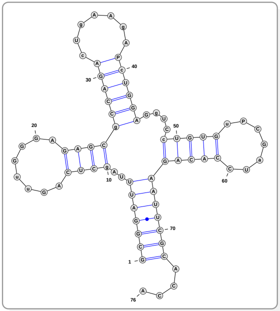
Non-crossing arc diagrams*

Supporting intuitions

Different representations

Common combinatorial structure

*Additional steric constraints

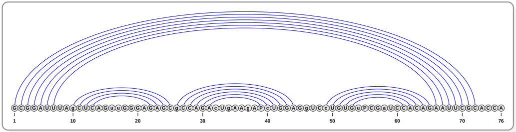


Outer-planar graphs

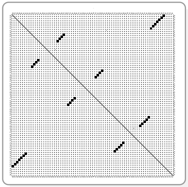
Hamiltonian-path, $\Delta(G) \leq 3$, 2-connected*

((((((((..((((........))))((((((.......))))))....(((((.......))))))))))))....

Motzkin words*

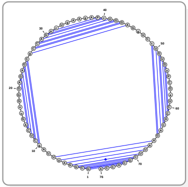


Non-crossing arc-annotated sequences*



Dot plots

Adjacency matrices*



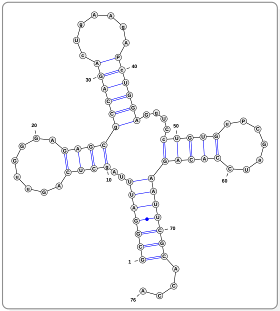
Non-crossing arc diagrams*

Supporting intuitions

Different representations

Common combinatorial structure

*Additional steric constraints

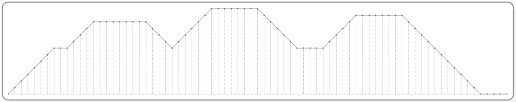


Outer-planar graphs

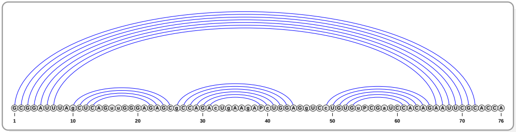
Hamiltonian-path, $\Delta(G) \leq 3$, 2-connected*

(((((((..((((........))))((((((.......))))))....(((((.......))))))))))))....

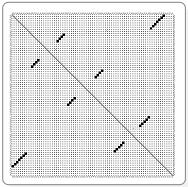
Motzkin words*



Positive 1D meanders* over $\mathcal{S} = \{+1, -1, 0\}$

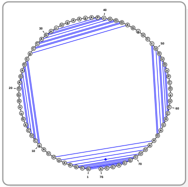


Non-crossing arc-annotated sequences*



Dot plots

Adjacency matrices*



Non-crossing arc diagrams*

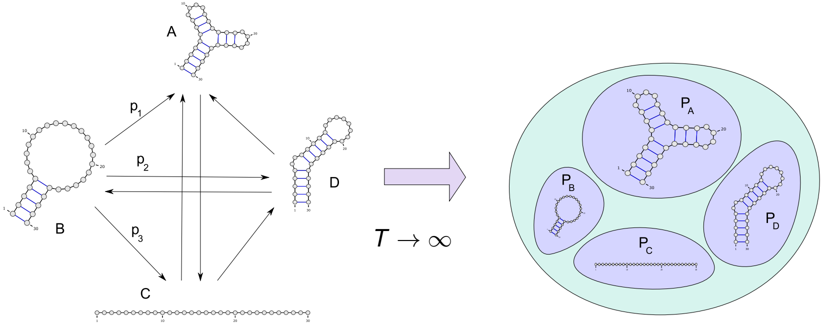
Supporting intuitions

Different representations

Common combinatorial structure

* Additional steric constraints

At the nanoscopic scale, RNA structure *fluctuates* ($\approx$ Markov process).

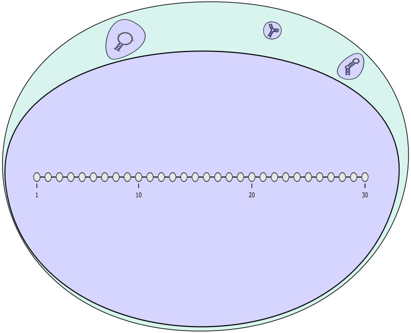


Convergence towards a stationary distribution at the Boltzmann equilibrium, where the probability of a conformation only depends on its free-energy.
Corollary: Initial conformation does not matter.

Questions: For a given **conformation space** and free-energy model:

- **A.** Determine most stable (Minimum Free-Energy) structure at equilibrium;
- **B.** Compute average properties of Boltzmann ensemble;

Transcription: RNA synthesized, supposedly without structure[2]

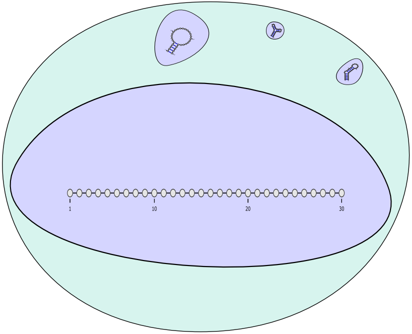


$T = 0$

But most mRNAs are degrade before 7h (Org.: Souris [SSN+09]).

[2] Except for co-transcriptional folding...

Transcription: RNA synthesized, supposedly without structure[2]

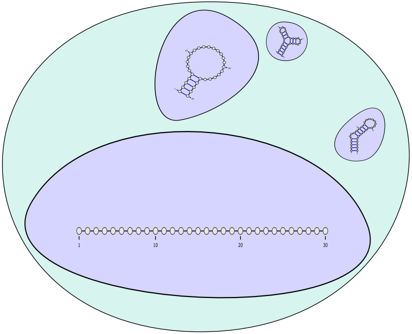


$T = 1h$

But most mRNAs are degrade before 7h (Org.: Souris [SSN+09]).

[2] Except for co-transcriptional folding...

Transcription: RNA synthesized, supposedly without structure[2]

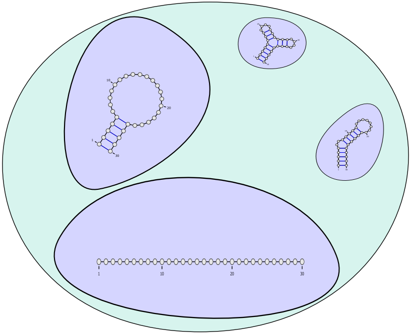


$T = 2h$

But most mRNAs are degrade before 7h (Org.: Souris [SSN+09]).

[2]Except for co-transcriptional folding. . .

Transcription: RNA synthesized, supposedly without structure[2]

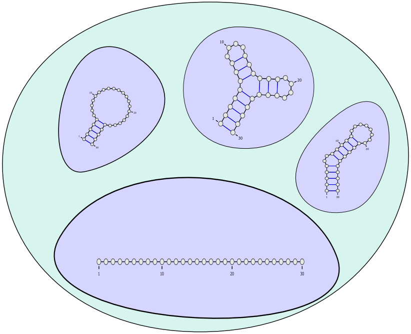


$T = 5h$

But most mRNAs are degrade before 7h (Org.: Souris [SSN+09]).

[2] Except for co-transcriptional folding...

Transcription: RNA synthesized, supposedly without structure[2]

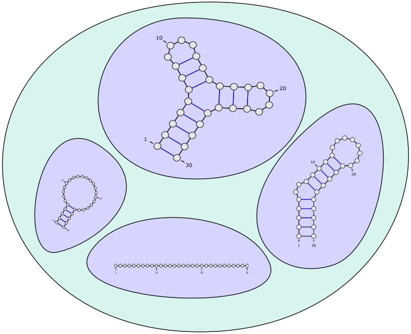


$T = 10h$

But most mRNAs are degrade before 7h (Org.: Souris [SSN+09]).

[2] Except for co-transcriptional folding...

Transcription: RNA synthesized, supposedly without structure[2]

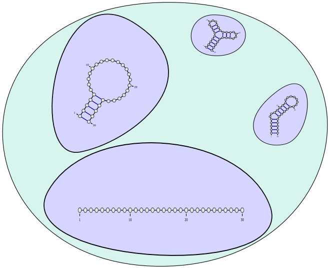


$T \to \infty$

But most mRNAs are degrade before 7h (Org.: Souris [SSN+09]).

[2] Except for co-transcriptional folding...

Transcription: RNA synthesized, supposedly without structure[2]



$T = 10h$

But most mRNAs are degrade before 7h (Org.: Souris [SSN$^+$09]).

A. Determine most stable (Minimum Free-Energy) structure at equilibrium;

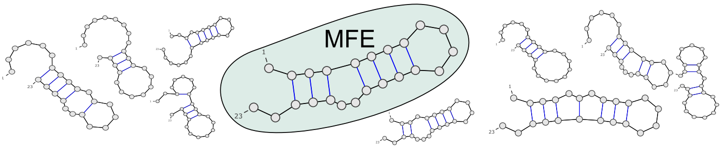
B. Compute average properties of Boltzmann ensemble;

C. **Determine most likely structure at finite time *T*.**
(c.f. H. Isambert through simulation, NP-complete deterministically [MTSC09])

[2] Except for co-transcriptional folding...

Problem A: Determine Minimum Free-Energy structure (MFE).

***Ab* initio folding prediction =**
Predict RNA structure from its sequence ω only.



- **Conformations:** Set S_ω of secondary structures compatible (w.r.t. base-pairing constraints) with primary structure ω .
- **Free-Energy:** Function $E_{\omega,S}$ (KCal.mol^{-1}), additive on motifs occurring in any sequence/conformation couple (ω, S).
- **Native structure:** Functional conformation of the biomolecule.

Remarks:

- Not necessarily unique (Kinetics, or bi-stable structures);
- In presence of PKs $\rightarrow$ Ambiguous: Which is the native conformation?

Nussinov/Jacobson energy model (NJ)

Base-pair maximization (with a twist):

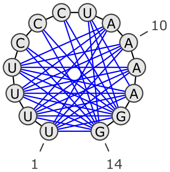
- Additive model on independently contributing base-pairs;
- Canonical base-pairs only: Watson/Crick (A/U,C/G) and Wobble (G/U)

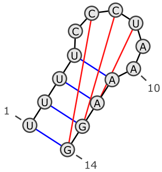
$$\Rightarrow E_{\omega,S} = -\#Paires(S)$$

Folding in NJ model ⇔ Base-pair (weight) maximization

Example:

UUUUCCCUAAAAGG





Variant: Weight each pair with −#Hydrogen bonds

$\Delta G(G{\equiv}C) = -3$ $\Delta G(A{=}U) = -2$ $\Delta G(G{-}U) = -1$

Nussinov/Jacobson energy model (NJ)

Base-pair maximization (with a twist):

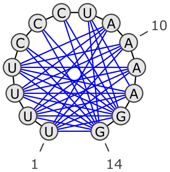
- Additive model on independently contributing base-pairs;
- Canonical base-pairs only: Watson/Crick (A/U,C/G) and Wobble (G/U)

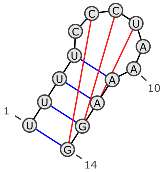
$$\Rightarrow E_{\omega,S} = -\#Paires(S)$$

Folding in NJ model $\Leftrightarrow$ Base-pair (weight) maximization

Example:

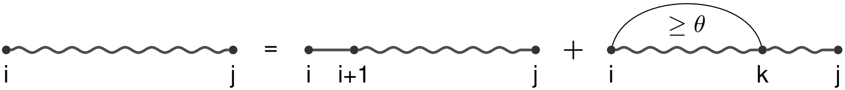
UUUUCCCUAAAAGG





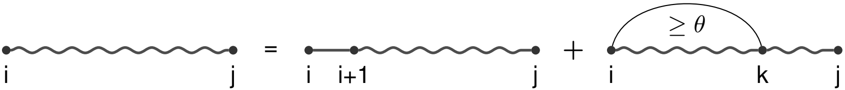
Variant: Weight each pair with −#Hydrogen bonds

ΔG(G≡C) = -3 ΔG(A=U) = -2 ΔG(G−U) = -1



$$N_{i,t} = 0, \quad \forall t \in [i, i+\theta]$$

$$N_{i,j} = \min \begin{cases} N_{i+1,j} & i \text{ unpaired} \\ \min\limits_{k=i+\theta+1}^{j} \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j} & i \text{ paired with } k \end{cases}$$

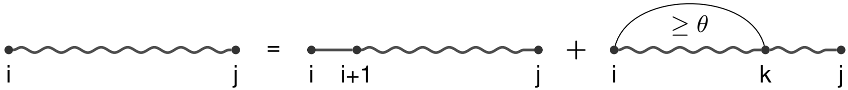


$$N_{i,t} = 0, \quad \forall t \in [i, i+\theta]$$

$$N_{i,j} = \min \begin{cases} N_{i+1,j} & i \text{ unpaired} \\ \min_{k=i+\theta+1}^{j} \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j} & i \text{ paired with } k \end{cases}$$

Correctness. Goal = Show that MFE over interval $[i, j]$ is indeed found in $N_{i,j}$ after completing the computation. Proceed by induction:

- Assume that property holds for any $[i', j']$ such that $j' - i' < n$.
- Consider $[i, j], j - i = n$. Let $\text{MFE}_{i,j}$:= Base-pairs of best struct. on $[i, j]$. Then first position i in $\text{MFE}_{i,j} =$ is either:
  - **Unpaired:** $\text{MFE}_{i,j} = \text{MFE}_{i+1,j}$ $\rightarrow$ free-energy $= N_{i+1,j}$
  - **Paired to k:** $\text{MFE}_{i,j} = \{(i,k)\} \cup \text{MFE}_{i+1,k-1} \cup \text{MFE}_{k+1,j}$.
    (Indeed, any BP between $[i+1, k-1]$ and $[k+1, j]$ would cross (i, k))
    $\rightarrow$ free-energy $= \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j}$

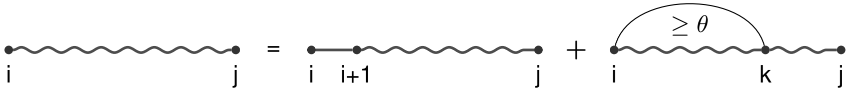


$$N_{i,t} = 0, \quad \forall t \in [i, i+\theta]$$

$$N_{i,j} = \min \begin{cases} N_{i+1,j} & i \text{ unpaired} \\ \min_{k=i+\theta+1}^{j} \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j} & i \text{ paired with } k \end{cases}$$

Correctness. Goal = Show that MFE over interval $[i,j]$ is indeed found in $N_{i,j}$ after completing the computation. Proceed by induction:

- Assume that property holds for any $[i',j']$ such that $j' - i' < n$.
- Consider $[i,j], j-i=n$. Let $\text{MFE}_{i,j}$:= Base-pairs of best struct. on $[i,j]$. Then first position i in $\text{MFE}_{i,j}$ = is either:
  - **Unpaired:** $\text{MFE}_{i,j} = \text{MFE}_{i+1,j}$ → free-energy $= N_{i+1,j}$
  - **Paired to k:** $\text{MFE}_{i,j} = \{(i,k)\} \cup \text{MFE}_{i+1,k-1} \cup \text{MFE}_{k+1,j}$.
    (Indeed, any BP between $[i+1,k-1]$ and $[k+1,j]$ would cross (i,k))
    → free-energy $= \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j}$



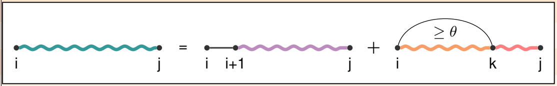
$$N_{i,t} = 0, \quad \forall t \in [i, i+\theta]$$

$$N_{i,j} = \min \begin{cases} N_{i+1,j} & i \text{ unpaired} \\ \min_{k=i+\theta+1}^{j} \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j} & i \text{ paired with } k \end{cases}$$

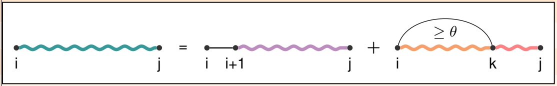
Correctness. Goal = Show that MFE over interval $[i, j]$ is indeed found in $N_{i,j}$ after completing the computation. Proceed by induction:

- Assume that property holds for any $[i', j']$ such that $j' - i' < n$.
- Consider $[i, j], j - i = n$. Let $\text{MFE}_{i,j}$:= Base-pairs of best struct. on $[i, j]$. Then first position i in $\text{MFE}_{i,j}$ = is either:
  - **Unpaired:** $\text{MFE}_{i,j} = \text{MFE}_{i+1,j}$ → free-energy $= N_{i+1,j}$
  - **Paired to k:** $\text{MFE}_{i,j} = \{(i,k)\} \cup \text{MFE}_{i+1,k-1} \cup \text{MFE}_{k+1,j}$.
    (Indeed, any BP between $[i+1, k-1]$ and $[k+1, j]$ would cross (i, k))
    → free-energy $= \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j}$

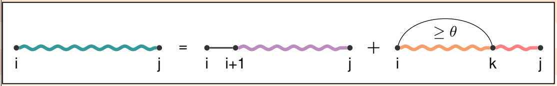
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



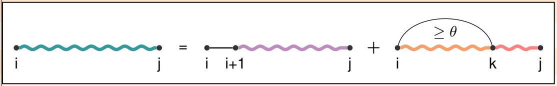
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



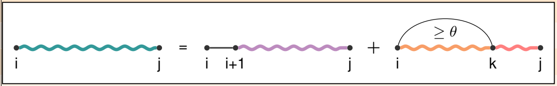
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



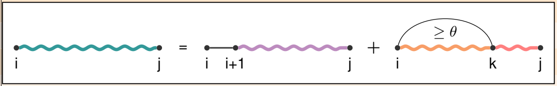
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



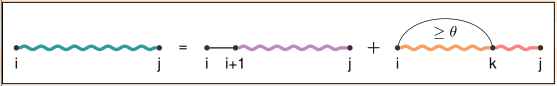
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



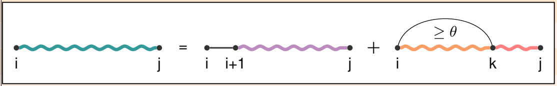
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



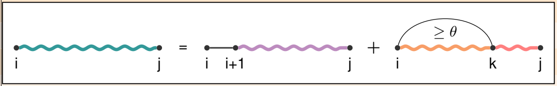
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



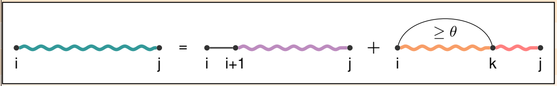
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



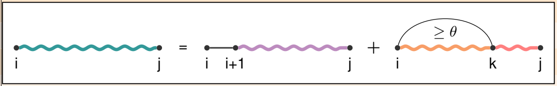
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



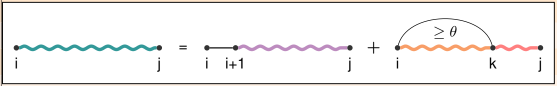
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



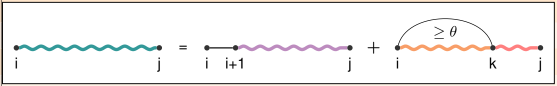
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



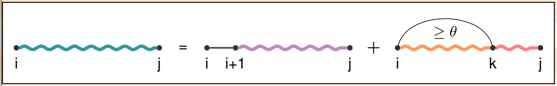
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



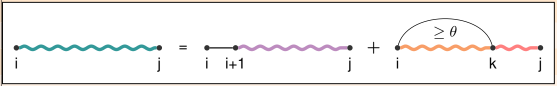
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



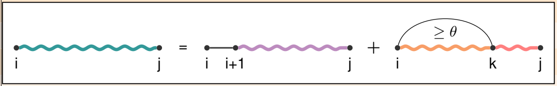
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	.	.	.	.	.	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



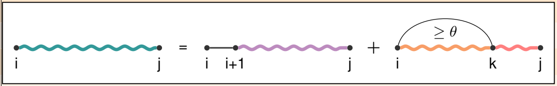
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	.	.	.	.	.	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



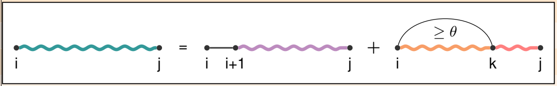
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	.	.	.	.	.	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



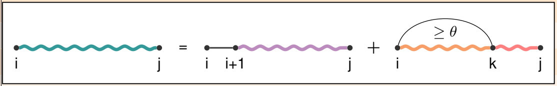
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



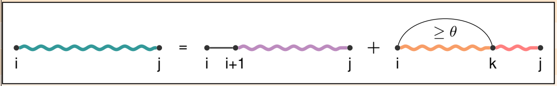
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



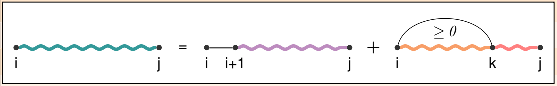
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



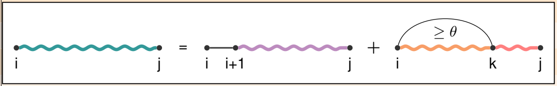
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



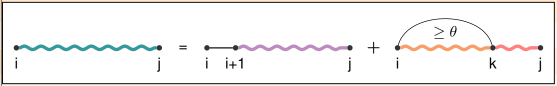
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



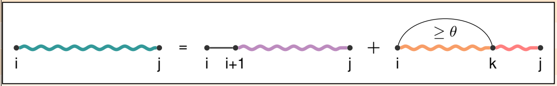
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



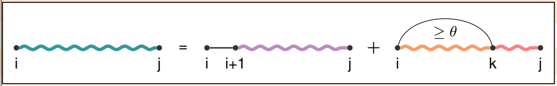
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	.	.	.	.	.	.	.	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



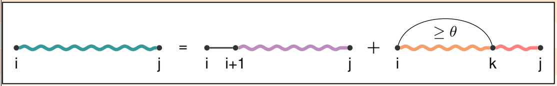
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	(	.	.	.	.	.	)	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



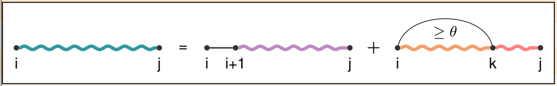
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	(	.	.	.	.	.	)	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



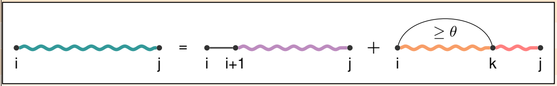
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	(	.	.	.	.	.	)	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	(	(	.	.	.	)	)	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



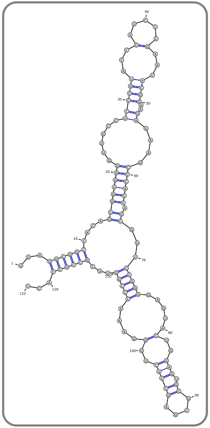
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	(	(	.	.	.	)	.	(	(	.	.	.	)	)	)	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	0	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0



Turner energy model

Based on unambiguous decomposition of 2^{ary} structure into loops:

- Internal loops
- Bulges
- Terminal loops
- Multi loops
- Stackings



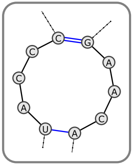
Free-energy Δ G of a loop depend on bases, assymmetry, dangles . . .

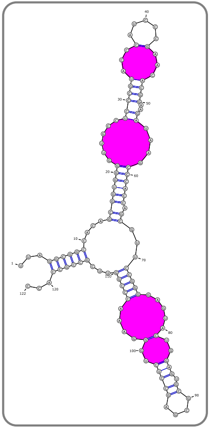
Experimentally determined
+ Interpolated for larger loops.

Improved results by taking stacking into account.

Based on unambiguous decomposition of 2^{ary} structure into loops:

- Internal loops
- Bulges
- Terminal loops
- Multi loops
- Stackings





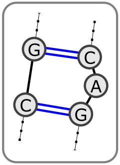
Free-energy Δ G of a loop depend on bases, assymmetry, dangles . . .

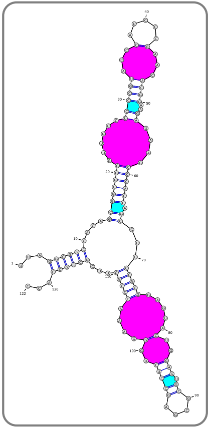
Experimentally determined
+ Interpolated for larger loops.

Improved results by taking stacking into account.

Based on unambiguous decomposition of 2^{ary} structure into loops:

- Internal loops
- Bulges
- Terminal loops
- Multi loops
- Stackings





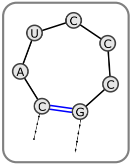
Free-energy Δ G of a loop depend on bases, assymmetry, dangles . . .

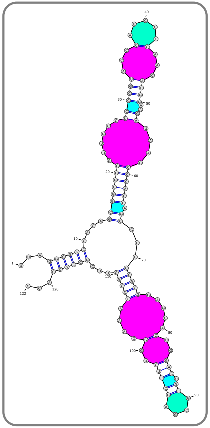
Experimentally determined
+ Interpolated for larger loops.

Improved results by taking stacking into account.

Based on unambiguous decomposition of 2^{ary} structure into loops:

- Internal loops
- Bulges
- Terminal loops
- Multi loops
- Stackings





Free-energy Δ G of a loop depend on bases, assymmetry, dangles . . .

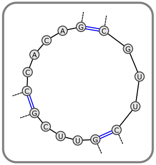
Experimentally determined
+ Interpolated for larger loops.

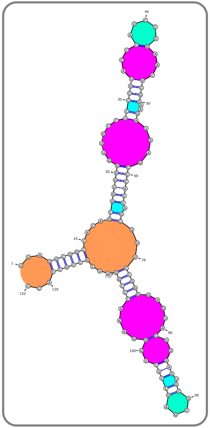
Improved results by taking stacking into account.

Turner energy model

Based on unambiguous decomposition of 2ary structure into loops:

- Internal loops
- Bulges
- Terminal loops
- Multi loops
- Stackings





Free-energy Δ G of a loop depend on bases, assymmetry, dangles . . .

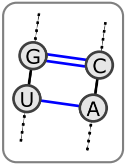
Experimentally determined
+ Interpolated for larger loops.

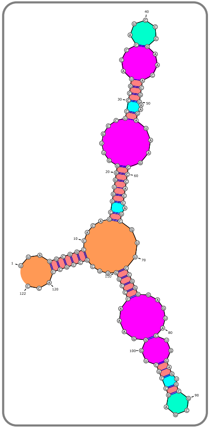
Improved results by taking stacking into account.

Turner energy model

Based on unambiguous decomposition of 2^{ary} structure into loops:

- Internal loops
- Bulges
- Terminal loops
- Multi loops
- Stackings

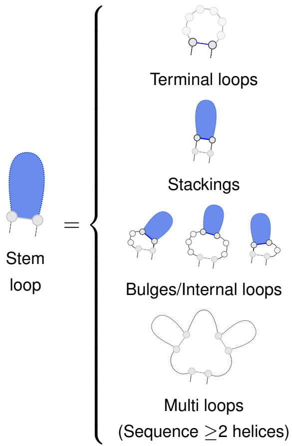




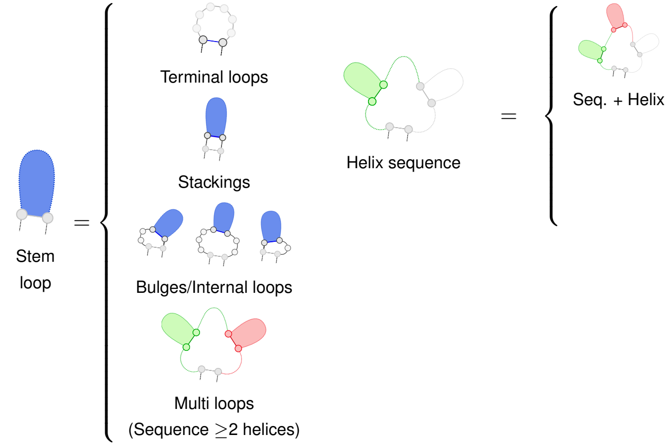
Free-energy Δ G of a loop depend on bases, assymmetry, dangles . . .

Experimentally determined
+ Interpolated for larger loops.

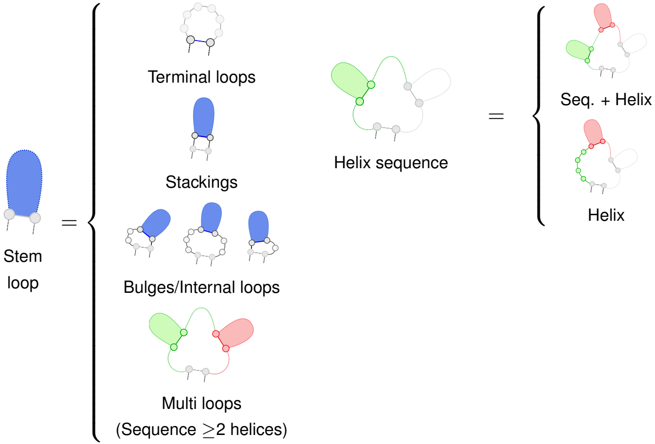
Improved results by taking stacking into account.


Stem
loop
=
Terminal loops
Stackings
Bulges/Internal loops
Multi loops
(Sequence ≥2 helices)

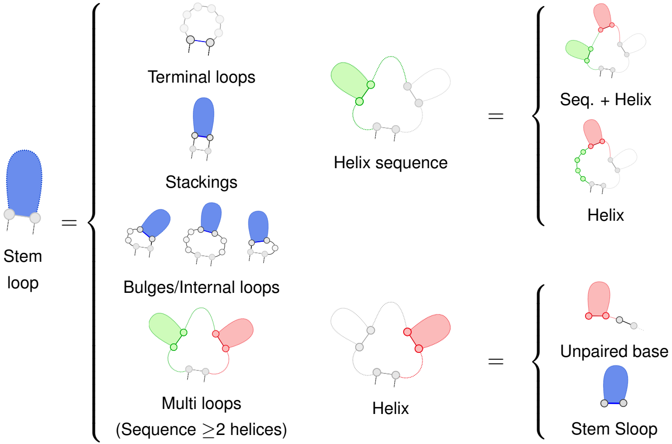
MFE DP equations



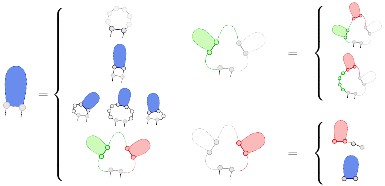
MFE DP equations



MFE DP equations



- $E_H(i,j)$: Energy of terminal loop *enclosed by* (i,j) pair
- $E_{BI}(i,j)$: Energy of bulge or internal loop *enclosed by* (i,j) pair
- $E_S(i,j)$: Energy of stacking $(i,j)/(i+1,j-1)$
- Penalty for multi loop (a), and occurrences of unpaired base (b) and helix (c) in multi loops.



DP recurrence

$$
\begin{aligned}
\mathcal{M}'_{i,j} &= \min \begin{cases} E_H(i,j) \\ E_S(i,j) + \mathcal{M}'_{i+1,j-1} \\ \text{Min}_{i',j'}\left(E_{BI}(i,i',j',j) + \mathcal{M}'_{i',j'}\right) \\ a + \text{Min}_k\left(\mathcal{M}_{i+1,k-1} + \mathcal{M}^1_{k,j-1}\right) \end{cases} \\
\mathcal{M}_{i,j} &= \text{Min}_k \left\{ \min\left(\mathcal{M}_{i,k-1}, b(k-1)\right) + \mathcal{M}^1_{k,j} \right\} \\
\mathcal{M}^1_{i,j} &= \text{Min}_k \left\{ b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j} \right\}
\end{aligned}
$$

Backtracking to reconstruct MFE structure:

$$\mathcal{M}'_{i,j} = \text{Min}\left\{\begin{array}{c} E_H(i,j) \\ E_S(i,j) + \mathcal{M}'_{i+1,j-1} \\ \text{Min}_{i',j'}(E_{BI}(i,i',j',j) + \mathcal{M}'_{i',j'}) \\ a + \text{Min}_k(\ \mathcal{M}_{i+1,k-1} + \mathcal{M}^1{}_{k,j-1}\ ) \end{array}\right\}$$

$$\mathcal{M}_{i,j} = \text{Min}_k\left\{\min(\mathcal{M}_{i,k-1}, b(k-1)) + \mathcal{M}^1{}_{k,j}\right\}$$

$$\mathcal{M}^1{}_{i,j} = \text{Min}_k\left\{b + \mathcal{M}^1{}_{i,j-1}, c + \mathcal{M}'_{i,j}\right\}$$

Complexity:

For each min, $\mathcal{O}(n)$ potential contributors
⇒ **Worst-case** complexity in $\mathcal{O}(n^2)$ for **naive backtrack**.
Keep best contributor for each Min ⇒ **Backtracking in $\mathcal{O}(n)$**

⇒ `UnaFold` [MZ08]/`RNAFold` [HFS+94] compute the MFE for the Turner model in **overall**[3] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

[3] Using a trick/restriction for internal loops...

Backtracking to reconstruct MFE structure:

$$\mathcal{M}'_{i,j} = \text{Min}\begin{Bmatrix} E_H(i,j) \\ E_S(i,j) + \mathcal{M}'_{i+1,j-1} \\ \text{Min}_{i',j'}\left(E_{BI}(i,i',j',j) + \mathcal{M}'_{i',j'}\right) \\ a + \text{Min}_k\left(\mathcal{M}_{i+1,k-1} + \mathcal{M}^1_{k,j-1}\right) \end{Bmatrix}$$

$$\mathcal{M}_{i,j} = \text{Min}_k\left\{\min\left(\mathcal{M}_{i,k-1}, b(k-1)\right) + \mathcal{M}^1_{k,j}\right\}$$

$$\mathcal{M}^1_{i,j} = \text{Min}_k\left\{b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j}\right\}$$

Complexity:

For each min, $\mathcal{O}(n)$ potential contributors
⇒ **Worst-case** complexity in $\mathcal{O}(n^2)$ for **naive backtrack**.
Keep best contributor for each Min ⇒ **Backtracking in $\mathcal{O}(n)$**

⇒ `UnaFold` [MZ08]/`RNAFold` [HFS+94] compute the MFE for the Turner model in **overall**[3] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

[3] Using a trick/restriction for internal loops...

Backtracking to reconstruct MFE structure:

???

$$
\begin{aligned}
\mathcal{M}'_{i,j} &= \text{Min}\left\{\begin{array}{l} E_H(i,j) \\ E_S(i,j) + \mathcal{M}'_{i+1,j-1} \\ \text{Min}_{i',j'}\left(E_{BI}(i,i',j',j) + \mathcal{M}'_{i',j'}\right) \\ a + \text{Min}_k\left(\mathcal{M}_{i+1,k-1} + \mathcal{M}^1_{k,j-1}\right) \end{array}\right\} \\
\mathcal{M}_{i,j} &= \text{Min}_k\left\{\min\left(\mathcal{M}_{i,k-1}, b(k-1)\right) + \mathcal{M}^1_{k,j}\right\} \\
\mathcal{M}^1_{i,j} &= \text{Min}_k\left\{b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j}\right\}
\end{aligned}
$$

Complexity:

For each min, $\mathcal{O}(n)$ potential contributors
$\Rightarrow$ **Worst-case** complexity in $\mathcal{O}(n^2)$ for **naive backtrack**.
Keep best contributor for each Min $\Rightarrow$ **Backtracking in $\mathcal{O}(n)$**

$\Rightarrow$ `UnaFold` [MZ08]/`RNAFold` [HFS+94] compute the MFE for the Turner model in **overall**[3] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

[3] Using a trick/restriction for internal loops...

Backtracking to reconstruct MFE structure:

$$\mathcal{M}'_{i,j} = \text{Min}\begin{Bmatrix} E_H(i,j) \\ E_S(i,j) + \mathcal{M}'_{i+1,j-1} \\ \text{Min}_{i',j'}\left(E_{BI}(i,i',j',j) + \mathcal{M}'_{i',j'}\right) \\ a + \text{Min}_k\left(\mathcal{M}_{i+1,k-1} + \mathcal{M}^1_{k,j-1}\right) \end{Bmatrix}$$

$$\mathcal{M}_{i,j} = \text{Min}_k\left\{\min\left(\mathcal{M}_{i,k-1}, b(k-1)\right) + \mathcal{M}^1_{k,j}\right\}$$

$$\mathcal{M}^1_{i,j} = \text{Min}_k\left\{b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j}\right\}$$

Complexity:

For each min, $\mathcal{O}(n)$ potential contributors
$\Rightarrow$ **Worst-case** complexity in $\mathcal{O}(n^2)$ for **naive backtrack**.
Keep best contributor for each Min $\Rightarrow$ **Backtracking in $\mathcal{O}(n)$**

$\Rightarrow$ `UnaFold` [MZ08]/`RNAFold` [HFS+94] compute the MFE for the Turner model in **overall**[3] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

[3] Using a trick/restriction for internal loops...

Backtracking to reconstruct MFE structure:

$$\mathcal{M}'_{i,j} = \text{Min}\left\{\begin{array}{c} E_H(i,j) \\ E_S(i,j) + \mathcal{M}'_{i+1,j-1} \\ \text{Min}_{i',j'}(E_{BI}(i,i',j',j) + \mathcal{M}'_{i',j'}) \\ a + \text{Min}_k(\mathcal{M}_{i+1,k-1} + \mathcal{M}^1_{k,j-1}) \end{array}\right\}$$

$$\mathcal{M}_{i,j} = \text{Min}_k\left\{\min(\mathcal{M}_{i,k-1}, b(k-1)) + \mathcal{M}^1_{k,j}\right\}$$

$$\mathcal{M}^1_{i,j} = \text{Min}_k\left\{b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j}\right\}$$

Complexity:

For each min, $\mathcal{O}(n)$ potential contributors
⇒ **Worst-case** complexity in $\mathcal{O}(n^2)$ for **naive backtrack**.
Keep best contributor for each Min ⇒ **Backtracking in $\mathcal{O}(n)$**

⇒ `UnaFold` [MZ08]/`RNAFold` [HFS+94] compute the MFE for the Turner model in **overall**[3] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

[3] Using a trick/restriction for internal loops...

Backtracking to reconstruct MFE structure:

$$\mathcal{M}'_{i,j} = \text{Min}\left\{\begin{array}{c} E_H(i,j) \\ E_S(i,j) + \mathcal{M}'_{i+1,j-1} \\ \text{Min}_{i',j'}(E_{BI}(i,i',j',j) + \mathcal{M}'_{i',j'}) \\ a + \text{Min}_k(\mathcal{M}_{i+1,k-1} + \mathcal{M}^1_{k,j-1}) \end{array}\right\}$$

$$\mathcal{M}_{i,j} = \text{Min}_k\left\{\min(\mathcal{M}_{i,k-1}, b(k-1)) + \mathcal{M}^1_{k,j}\right\}$$

$$\mathcal{M}^1_{i,j} = \text{Min}_k\left\{b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j}\right\}$$

Complexity:

For each min, $\mathcal{O}(n)$ potential contributors
⇒ **Worst-case** complexity in $\mathcal{O}(n^2)$ for **naive backtrack**.
Keep best contributor for each Min ⇒ **Backtracking in $\mathcal{O}(n)$**

⇒ UnaFold [MZ08]/RNAFold [HFS+94] compute the MFE for the Turner model in **overall**[3] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

[3] Using a trick/restriction for internal loops...

Backtracking to reconstruct MFE structure:

$$\mathcal{M}'_{i,j} = \text{Min}\left\{\begin{array}{c} E_H(i,j) \\ E_S(i,j) + \mathcal{M}'_{i+1,j-1} \\ \text{Min}_{i',j'}(E_{BI}(i,i',j',j) + \mathcal{M}'_{i',j'}) \\ a + \text{Min}_k(\mathcal{M}_{i+1,k-1} + \mathcal{M}^1_{k,j-1}) \end{array}\right\}$$

$$\mathcal{M}_{i,j} = \text{Min}_k\left\{\min(\mathcal{M}_{i,k-1}, b(k-1)) + \mathcal{M}^1_{k,j}\right\}$$

$$\mathcal{M}^1_{i,j} = \text{Min}_k\left\{b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j}\right\}$$

Complexity:

For each min, $\mathcal{O}(n)$ potential contributors
⇒ **Worst-case** complexity in $\mathcal{O}(n^2)$ for **naive backtrack**.
Keep best contributor for each Min ⇒ **Backtracking in $\mathcal{O}(n)$**

⇒ `UnaFold` [MZ08]/`RNAFold` [HFS+94] compute the MFE for the Turner model in **overall**[3] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

[3] Using a trick/restriction for internal loops...

Definition (Ab initio folding)

Starting from sequence, find conformation that minimizes free-energy.

Advantages:

- Mechanical nature allows the (in)validation of models
- Reasonable complexity $\mathcal{O}(n^3)/\mathcal{O}(n^2)$ time/space
- *Exhaustive* nature

Limitations:

- Hard to include PKs
- Highly dependent on energy model
- No cooperativity
- Limited performances

Definition (Comparative approach)

Starting from homologous sequences, postulate common structure and find best possible tradeoff between folding & alignment.

Avantages :

- Better performances
- (Limited) cooperativity
- Self-improving

Limitations

- Easily unreasonable complexity
- Non exhaustive search
- Captures *transient* structures

Definition (Ab initio folding)

Starting from sequence, find conformation that minimizes free-energy.

Advantages:

- Mechanical nature allows the (in)validation of models
- Reasonable complexity $\mathcal{O}(n^3)/\mathcal{O}(n^2)$ time/space
- *Exhaustive* nature

Limitations:

- Hard to include PKs
- Highly dependent on energy model
- No cooperativity
- Limited performances

Definition (Comparative approach)

Starting from homologous sequences, postulate common structure and find best possible tradeoff between folding & alignment.

Avantages :

- Better performances
- (Limited) cooperativity
- Self-improving

Limitations

- Easily unreasonable complexity
- Non exhaustive search
- Captures *transient* structures

Taille	Sens.
<700	70-73%

[MSZT99, DCCG04]

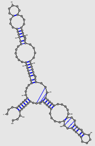
16s,23s	$\sim$50%

MCC$\sim$ 0.5 [GG04]

Thermodynamics

UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA
CC

Sequence



2^{ary} Structure

Sens.	Spé.	MCC.
$\sim$75%	$\sim$75%	0.8

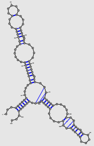
Comparative [GG04]

Reminder: $MCC = \frac{t^+ t^- - f^+ f^-}{\sqrt{(t^+ + f^+)(t^+ + f^-)(t^- + f^+)(t^- + f^-)}}$

Taille Sens.
<700 70-73%
[MSZT99, DCCG04]
16s,23s ∼50%
MCC∼ 0.5 [GG04]
Thermodynamics

UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA
CC

Sequence



2ary Structure

Sens. Spé. MCC.
∼75% ∼75% 0.8
Comparative [GG04]

Reminder: $MCC = \frac{t^+ t^- - f^+ f^-}{\sqrt{(t^+ + f^+)(t^+ + f^-)(t^- + f^+)(t^- + f^-)}}$

Performances

UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA
CC

Sequence

Taille	Sens.
<700	70-73%

[MSZT99, DCCG04]

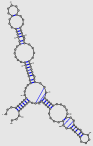
16s,23s	$\sim$50%

MCC$\sim$ 0.5 [GG04]

Thermodynamics

Sens.	Spé.	MCC.
$\sim$75%	$\sim$75%	0.8

Comparative [GG04]



2^{ary} Structure

Reminder: $MCC = \frac{t^+ t^- - f^+ f^-}{\sqrt{(t^+ + f^+)(t^+ + f^-)(t^- + f^+)(t^- + f^-)}}$

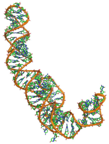
Towards a 3D ab-initio prediction

Goal: From sequence to all-atom/coarse grain 3D models!!!

- Comparative models + Molecular dynamics: `RNA2D3D` [SYKB07]
- Pipeline `MC-Fold/MC-sym` [PM08]

```
UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA
CC
```

Séquence



3^{ary} Structure

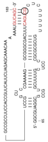
Goal: From sequence to all-atom/coarse grain 3D models!!!

- Comparative models + Molecular dynamics: `RNA2D3D` [SYKB07]
- Pipeline `MC-Fold/MC-sym` [PM08]

```
UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA
CC
```

Séquence

Comparative
`KNetFold`



2^{ary} Structure
+ Pseudoknots

Molecular
dynamics

3^{ary} Structure

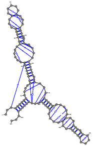
Goal: From sequence to all-atom/coarse grain 3D models!!!

- Comparative models + Molecular dynamics: `RNA2D3D` [SYKB07]
- Pipeline `MC-Fold/MC-sym` [PM08]

```
UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA
CC
```

Séquence

→ `MCFold` →



2^{ary+} Structure

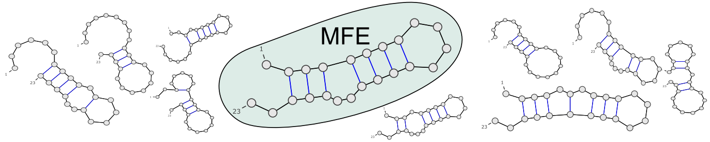
→ `MCSym` →

3^{ary} Structure

RNA *breathes* ⇒ There is no more than a single conformation.

New paradigm

The conformations of an RNA **coexist** in the the **Boltzmann distribution**.



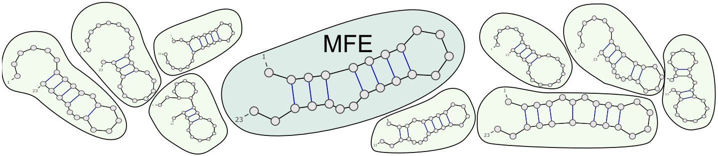
Consequence: The MFE probability can be arbitrarily small.
⇒ To understand how RNA acts, one must account for the set of alternative structures.

In particular, structurally close structures may *ally*, and become the most realistic candidate in the search for a functional conformation.

RNA *breathes* ⇒ There is no more than a single conformation.

New paradigm

The conformations of an RNA **coexist** in the **Boltzmann distribution**.



Consequence: The MFE probability can be arbitrarily small.
⇒ To understand how RNA acts, one must account for the set of alternative structures.

In particular, structurally close structures may *ally*, and become the most realistic candidate in the search for a functional conformation.

For each structure S compatible with an RNA ω, the Boltzmann distribution associates a **Boltzmann factor** $\mathcal{B}_{S,\omega} = e^{\frac{-E_{S,\omega}}{RT}}$, where:

- $E_{S,\omega}$ is the free-energy S (kCal.mol^{-1})
- T is the temperature (K)
- R is the perfect gaz constant (1.986.10^{-3} kCal.K^{-1}.mol^{-1})

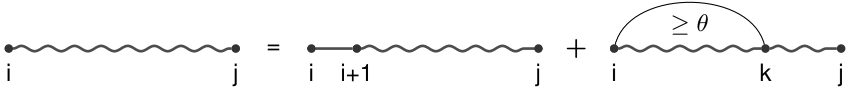
To obtain a distribution, one simply renormalizes by the **partition function**

$$\mathcal{Z}_\omega = \sum_{S \in \mathcal{S}_\omega} e^{\frac{-E_{S,\omega}}{RT}}$$

where $\mathcal{S}_\omega$ is the set of conformations that are compatibles with ω.

The **Boltzmann probability** of a structure S is simply given by

$$P_{S,\omega} = \frac{e^{\frac{-E_{S,\omega}}{RT}}}{\mathcal{Z}_\omega}.$$

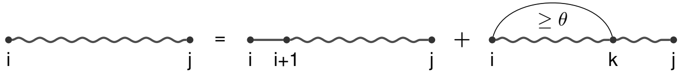


$$N_{i,t} = 0, \quad \forall t \in [i, i+\theta]$$

$$N_{i,j} = \min \begin{cases} N_{i+1,j} & i \text{ unpaired} \\ \min_{k=i+\theta+1}^{j} \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j} & i \text{ paired with } k \end{cases}$$

Ambiguity? Consider i: Either **unpaired**, or **paired** to k.
Sets of structures generated in these two cases are clearly disjoint.
(also holds for various values of k) $\Rightarrow$ **Unambiguous** decomposition

Completeness?True, since scheme explores every possible outcome for i.
+ Induction on interval length $\Rightarrow$ **Complete** decomposition



Recurrence for minimal free-energy of a fold :

$$N_{i,t} = 0, \quad \forall t \in [i, i+\theta]$$

$$N_{i,j} = \min \begin{cases} N_{i+1,j} & (i \text{ unpaired}) \\ \min_{k=i+\theta+1}^{j} E_{i,k} + N_{i+1,k-1} + N_{k+1,j} & (i \text{ comp. with } k) \end{cases}$$

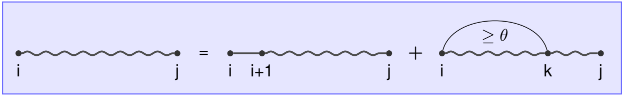
Recurrence for counting compatible structures :

$$C_{i,t} = 1, \quad \forall t \in [i, i+\theta]$$

$$C_{i,j} = \sum \begin{cases} C_{i+1,j} & (i \text{ unpaired}) \\ \sum_{k=i+\theta+1}^{j} 1 \times C_{i+1,k-1} \times C_{k+1,j} & (i \text{ comp. with } k) \end{cases}$$

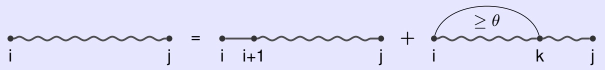
Decomposition matters, and the rest (MFE, count. . .) follows!

Partition function = **Weighted count** over compatible structures



$$
\begin{aligned}
\mathcal{Z}_{i,t} &= 1, \quad \forall t \in [i, i+\theta] \\
\mathcal{Z}_{i,j} &= \sum \begin{cases} \mathcal{Z}_{i+1,j} \\ \sum_{k=i+\theta+1}^{j} 1 \times \mathcal{Z}_{i+1,k-1} \times \mathcal{Z}_{k+1,j} \end{cases}
\end{aligned}
$$

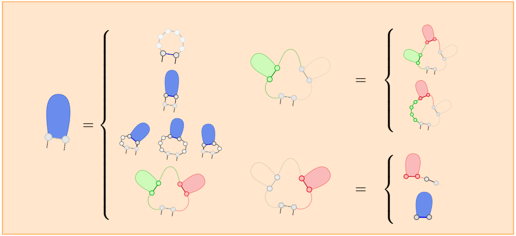
Partition function = **Weighted count** over compatible structures



$$\mathcal{Z}_{i,t} = 1, \quad \forall t \in [i, i+\theta]$$

$$\mathcal{Z}_{i,j} = \sum \begin{cases} \mathcal{Z}_{i+1,j} \\ \displaystyle\sum_{k=i+\theta+1}^{j} e^{\frac{-E_{\text{bp}}(i,k)}{RT}} \times \mathcal{Z}_{i+1,k-1} \times \mathcal{Z}_{k+1,j} \end{cases}$$

Partition function = **Weighted count** over compatible structures

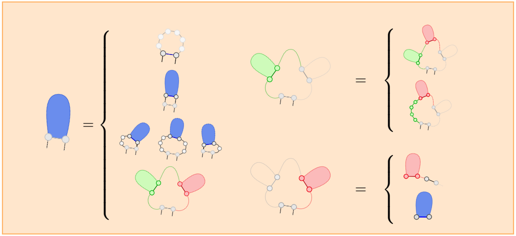


$$\mathcal{M}'_{i,j} = \text{Min}\begin{cases} E_H(i,j) \\ E_S(i,j) + \mathcal{M}'_{i+1,j-1} \\ \text{Min}(E_{BI}(i,i',j',j) + \mathcal{M}'_{i',j'}) \\ a + \text{Min}\left(\mathcal{M}_{i+1,k-1} + \mathcal{M}^1_{k,j-1}\right) \end{cases}$$

$$\mathcal{M}_{i,j} = \text{Min}\left\{\text{Min}\left(\mathcal{M}_{i,k-1}, b(k-1)\right) + \mathcal{M}^1_{k,j}\right\}$$

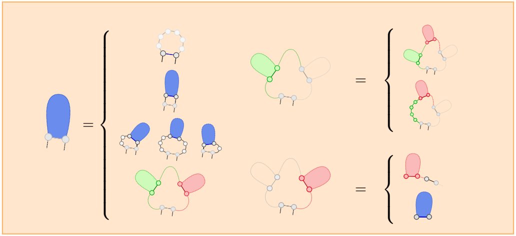
$$\mathcal{M}^1_{i,j} = \text{Min}\left\{b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j}\right\}$$

Partition function = **Weighted count** over compatible structures



$$
\begin{aligned}
\mathcal{M}'_{i,j} &= \text{Min}\begin{cases} e^{\frac{-E_H(i,j)}{RT}} \\ e^{\frac{-E_S(i,j)}{RT}} + \mathcal{M}'_{i+1,j-1} \\ \text{Min}\left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} + \mathcal{M}'_{i',j'}\right) \\ e^{\frac{-(a)}{RT}} + \text{Min}\left(\mathcal{M}_{i+1,k-1} + \mathcal{M}^1{}_{k,j-1}\right) \end{cases} \\
\mathcal{M}_{i,j} &= \text{Min}\left\{\text{Min}\left(\mathcal{M}_{i,k-1}, e^{\frac{-b(k-1)}{RT}}\right) + \mathcal{M}^1{}_{k,j}\right\} \\
\mathcal{M}^1{}_{i,j} &= \text{Min}\left\{e^{\frac{-b}{RT}} + \mathcal{M}^1{}_{i,j-1}, e^{\frac{-c}{RT}} + \mathcal{M}'_{i,j}\right\}
\end{aligned}
$$

Partition function = **Weighted count** over compatible structures

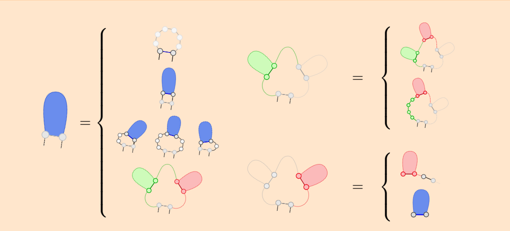


$$
\mathcal{M}'_{i,j} = \text{Min}\begin{cases} e^{\frac{-E_H(i,j)}{RT}} \\ e^{\frac{-E_S(i,j)}{RT}}\,\mathcal{M}'_{i+1,j-1} \\ \text{Min}\left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}}\,\mathcal{M}'_{i',j'}\right) \\ e^{\frac{-(a)}{RT}}\,\text{Min}\left(\mathcal{M}_{i+1,k-1}\mathcal{M}^1{}_{k,j-1}\right) \end{cases}
$$

$$
\mathcal{M}_{i,j} = \text{Min}\left\{\text{Min}\left(\mathcal{M}_{i,k-1}, e^{\frac{-b(k-1)}{RT}}\right)\mathcal{M}^1{}_{k,j}\right\}
$$

$$
\mathcal{M}^1{}_{i,j} = \text{Min}\left\{e^{\frac{-b}{RT}}\,\mathcal{M}^1{}_{i,j-1}, e^{\frac{-c}{RT}}\,\mathcal{M}'_{i,j}\right\}
$$

Partition function = **Weighted count** over compatible structures



$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} \\ e^{\frac{-E_S(i,j)}{RT}}\,\mathcal{Z}'(i+1,j-1) \\ +\sum\left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}}\,\mathcal{Z}'(i',j')\right) \\ +e^{\frac{-(a)}{RT}}\sum\left(\mathcal{Z}(i+1,k-1)\mathcal{Z}^1(k,j-1)\right) \end{cases}$$

$$\mathcal{Z}(i,j) = \sum\left(\mathcal{Z}(i,k-1)+e^{\frac{-b(k-1)}{RT}}\right)\mathcal{Z}^1(k,j)$$

$$\mathcal{Z}^1(i,j) = e^{\frac{-b}{RT}}\,\mathcal{Z}^1(i,j-1)+e^{\frac{-c}{RT}}\,\mathcal{Z}'(i,j)$$

Partition function = **Weighted count** over compatible structures

$$
\begin{aligned}
\mathcal{Z}_{i,t} &= 1, \quad \forall t \in [i, i+\theta] \\
\mathcal{Z}_{i,j} &= \sum \begin{cases} \mathcal{Z}_{i+1,j} \\ \displaystyle\sum_{k=i+\theta+1}^{j} e^{\frac{-E_{\text{bp}}(i,k)}{RT}} \times \mathcal{Z}_{i+1,k-1} \times \mathcal{Z}_{k+1,j} \end{cases}
\end{aligned}
$$

Validity of a partition function computation:

- Completeness/Unambiguity of decomposition scheme
- Correctness of Boltzmann factor
  Weight induced by backtrack = Product of derivations weights
  $e^{-E/RT} \to$ Weight products $\Leftrightarrow$ Summing energy terms

$$
\begin{aligned}
e^{-E_{\text{bp}}(i,k)/RT} \times \mathcal{Z}_{i+1,k-1} \times \mathcal{Z}_{k+1,j} &= \cdot \sum_x e^{-E(x)/RT} \cdot \sum_y e^{-E(y)/RT} \\
&= \sum_{x,y} e^{-a/RT} \cdot e^{-E(x)/RT} \cdot e^{-E(y)/RT} \\
&= \sum_{x,y} e^{-(E_{\text{bp}}(i,k)+E(x)+E(y))/RT}
\end{aligned}
$$

Partition function = **Weighted count** over compatible structures

$$\mathcal{Z}_{i,t} = 1, \quad \forall t \in [i, i+\theta]$$

$$\mathcal{Z}_{i,j} = \sum \begin{cases} \mathcal{Z}_{i+1,j} \\ \sum_{k=i+\theta+1}^{j} e^{\frac{-E_{\text{bp}}(i,k)}{RT}} \times \mathcal{Z}_{i+1,k-1} \times \mathcal{Z}_{k+1,j} \end{cases}$$

Validity of a partition function computation:

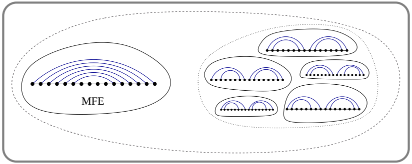
- Completeness/Unambiguity of decomposition scheme
- Correctness of Boltzmann factor
  Weight induced by backtrack = Product of derivations weights
  $e^{-E/RT}$ → Weight products ⇔ Summing energy terms

$$e^{-E_{\text{bp}}(i,k)/RT} \times \mathcal{Z}_{i+1,k-1} \times \mathcal{Z}_{k+1,j} = \cdot \sum_{x} e^{-E(x)/RT} \cdot \sum_{y} e^{-E(y)/RT}$$

$$= \sum_{x,y} e^{-a/RT} \cdot e^{-E(x)/RT} \cdot e^{-E(y)/RT}$$

$$= \sum_{x,y} e^{-(E_{\text{bp}}(i,k)+E(x)+E(y))/RT}$$

MFE ($\Leftrightarrow$ Max probability) may be **heavily dominated** by a set $\mathcal{B}$ of **structurally similar** suboptimal structures.
$\Rightarrow$ Functional conformation probably closer to $\mathcal{B}$ than to MFE.



Proof-of-concept: [DCL05]

- Sample structures within Boltzmann probability
- Cluster structures
- Build and return consensus structure of the heaviest cluster

$\Rightarrow$ Relative improvement for specificity (+17.6%) and sensitivity (+21.74%, except group II introns)

Problem

How to sample from the Boltzmann ensemble?

Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices ($\mathcal{Z}, \mathcal{Z}', \mathcal{Z}^1$).
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum ??? \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1,j-1) & \text{A} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{B} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1,k-1)\mathcal{Z}^1(k,j-1) \right) & \text{C} \end{cases}$$

Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices ($\mathcal{Z}, \mathcal{Z}', \mathcal{Z}^1$).
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1,j-1) & \text{(A)} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{(B)} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1,k-1)\mathcal{Z}^1(k,j-1) \right) & \text{(C)} \end{cases}$$

Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices ($\mathcal{Z}$, $\mathcal{Z}'$, $\mathcal{Z}^1$).
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1,j-1) & \text{(A)} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{(B)} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1,k-1)\mathcal{Z}^1(k,j-1) \right) & \text{(C)} \end{cases}$$

$r \downarrow$

$A_1 | A_2 | B_i | B_{i+1} | \ldots | B_{j-1} | B_j | C_i | C_{i+1} | \ldots | C_{j-1} | C_j$

Stochastic backtrack (adapted from `SFold`)

Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices $(\mathcal{Z}, \mathcal{Z}', \mathcal{Z}^1)$.
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1, j-1) & \text{(A)} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{(B)} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1, k-1) \mathcal{Z}^1(k, j-1) \right) & \text{(C)} \end{cases}$$

r ↓

$$A_1 | A_2 | B_i | B_{i+1} | \ldots | B_{j-1} | B_j | C_i | C_{i+1} | \ldots | C_{j-1} | C_j$$

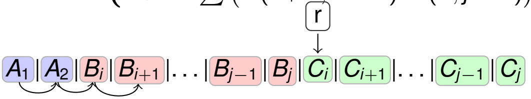
Stochastic backtrack (adapted from SFold)

Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices $(\mathcal{Z}, \mathcal{Z}', \mathcal{Z}^1)$.
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1, j-1) & \text{A} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{B} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1, k-1) \mathcal{Z}^1(k, j-1) \right) & \text{C} \end{cases}$$

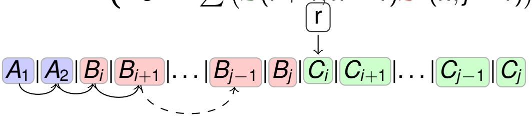


Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices ($\mathcal{Z}$, $\mathcal{Z}'$, $\mathcal{Z}^1$).
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1,j-1) & \text{A} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{B} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1,k-1) \mathcal{Z}^1(k,j-1) \right) & \text{C} \end{cases}$$

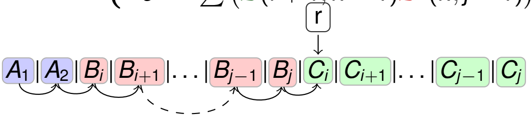


Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices $(\mathcal{Z}, \mathcal{Z}', \mathcal{Z}^1)$.
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1, j-1) & \text{A} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{B} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1, k-1) \mathcal{Z}^1(k, j-1) \right) & \text{C} \end{cases}$$



Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices $(\mathcal{Z}, \mathcal{Z}', \mathcal{Z}^1)$.
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1, j-1) & \text{A} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{B} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1, k-1) \mathcal{Z}^1(k, j-1) \right) & \text{C} \end{cases}$$

Correctness: Each $S \in \mathcal{S}_\omega$ uniquely generated (DP scheme unambiguity)
Therefore the probability of generated S is

$$p_S = \frac{\mathcal{B}(E_1)}{\mathcal{B}(\mathcal{S}_w)} \cdot \frac{\mathcal{B}(E_2)}{\mathcal{B}(E_1)} \cdot \frac{\mathcal{B}(E_3)}{\mathcal{B}(E_2)} \cdots \frac{\mathcal{B}(\{S\})}{\mathcal{B}(E_m)}$$

Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices ($\mathcal{Z}, \mathcal{Z}', \mathcal{Z}^1$).
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1,j-1) & \text{(A)} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{(B)} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1,k-1)\mathcal{Z}^1(k,j-1) \right) & \text{(C)} \end{cases}$$

Correctness: Each $S \in \mathcal{S}_\omega$ uniquely generated (DP scheme unambiguity)
Therefore the probability of generated S is

$$p_S = \frac{1}{\mathcal{B}(\mathcal{S}_w)} \cdot \frac{1}{1} \cdot \frac{1}{1} \ldots \frac{\mathcal{B}(\{S\})}{1}$$

Goal [DL03]**:** From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Principle: Choose derivation with prob. prop. to its contribution to part. fun.
Precomputation: Compute part. fun. versions of matrices ($\mathcal{Z}, \mathcal{Z}', \mathcal{Z}^1$).
Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1,j-1) & \text{A} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{B} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1,k-1)\mathcal{Z}^1(k,j-1) \right) & \text{C} \end{cases}$$

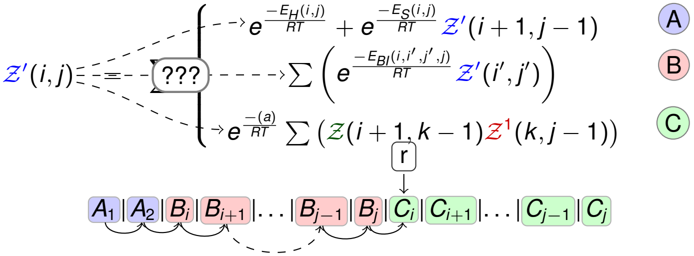
Correctness: Each $S \in \mathcal{S}_\omega$ uniquely generated (DP scheme unambiguity)
Therefore the probability of generated S is

$$p_S = \frac{\mathcal{B}(\{S\})}{\mathcal{B}(\mathcal{S}_w)} = \frac{e^{-E_s/RT}}{\mathcal{Z}} = P_{S,\omega}$$

Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices



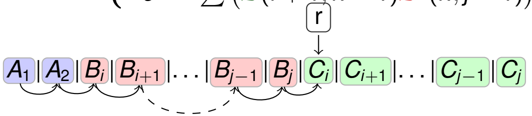
Average-case complexity in $\Theta(k \times n\sqrt{n})$ (homopolymer model) [Pon08].
Boustrophedon search $\Rightarrow \mathcal{O}(k \times n \log n)$ worst-case [Pon08].

Goal [DL03]: From sequence ω, draw S with prob. $e^{-E_S/RT}/\mathcal{Z}$

Stochastic backtrack:

1. Draw uniform random number $r \in [0, \mathcal{Z}'(i,j))$
2. Subtract from r the contributions of $\mathcal{Z}'(i,j)$ until $r < 0$
3. Recurse over associated regions/matrices

$$\mathcal{Z}'(i,j) = \sum \begin{cases} e^{\frac{-E_H(i,j)}{RT}} + e^{\frac{-E_S(i,j)}{RT}} \mathcal{Z}'(i+1,j-1) & \text{(A)} \\ \sum \left(e^{\frac{-E_{BI}(i,i',j',j)}{RT}} \mathcal{Z}'(i',j') \right) & \text{(B)} \\ e^{\frac{-(a)}{RT}} \sum \left(\mathcal{Z}(i+1,k-1)\mathcal{Z}^1(k,j-1) \right) & \text{(C)} \end{cases}$$



After $\Theta(n)$ operations, recurse over region of length $n-1$
$\Rightarrow$ Worst-case complexity in $\mathcal{O}(k \times n^2)$ for k samples

Average-case complexity in $\Theta(k \times n\sqrt{n})$ (homopolymer model) [Pon08].
Boustrophedon search $\Rightarrow \mathcal{O}(k \times n \log n)$ worst-case [Pon08].

A. Condon, B. Davy, B. Rastegari, S. Zhao, and F. Tarrant.
Classifying RNA pseudoknotted structures.
Theoretical Computer Science, 320(1):35–50, 2004.

K. Doshi, J. J. Cannone, C. Cobaugh, and R. R. Gutell.
Evaluation of the suitability of free-energy minimization using nearest-neighbor energy parameters for rna secondary structure prediction.
BMC Bioinformatics, 5(1):105, 2004.

Y. Ding, C. Y. Chan, and C. E. Lawrence.
RNA secondary structure prediction by centroids in a boltzmann weighted ensemble.
RNA, 11:1157–1166, 2005.

Y. Ding and E. Lawrence.
A statistical sampling algorithm for RNA secondary structure prediction.
Nucleic Acids Research, 31(24):7280–7301, 2003.

P. Gardner and R. Giegerich.
A comprehensive comparison of comparative rna structure prediction approaches.
BMC Bioinformatics, 5(1):140, 2004.

I. L. Hofacker, W. Fontana, P. F. Stadler, L. S. Bonhoeffer, M. Tacker, and P. Schuster.
Fast folding and comparison of RNA secondary structures.
Monatshefte für Chemie / Chemical Monthly, 125(2):167–188, 1994.

R. B. Lyngsøand C. N. S. Pedersen.
RNA pseudoknot prediction in energy-based models.
Journal of Computational Biology, 7(3-4):409–427, 2000.

N. Leontis and E. Westhof.
Geometric nomenclature and classification of RNA base pairs.
RNA, 7:499–512, 2001.

D.H. Mathews, J. Sabina, M. Zuker, and D.H. Turner.
Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure.
J Mol Biol, 288:911–940, 1999.

Jan Manuch, Chris Thachuk, Ladislav Stacho, and Anne Condon.
Np-completeness of the direct energy barrier problem without pseudoknots.
In Russell Deaton and Akira Suyama, editors, *DNA Computing and Molecular Programming*, volume 5877 of *Lecture Notes in Computer Science*, pages 106–115. Springer Berlin Heidelberg, 2009.

N. R. Markham and M. Zuker.
Bioinformatics, chapter UNAFold, pages 3–31.
Springer, 2008.

M. Parisien and F. Major.
The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data.
Nature, 452(7183):51–55, 2008.

Y. Ponty.
Efficient sampling of RNA secondary structures from the boltzmann ensemble of low-energy: The boustrophedon method.
Journal of Mathematical Biology, 56(1-2):107–127, Jan 2008.

Lioudmila V Sharova, Alexei A Sharov, Timur Nedorezov, Yulan Piao, Nabeebi Shaik, and Minoru S H Ko.
Database for mrna half-life of 19 977 genes obtained by dna microarray analysis of pluripotent and differentiating mouse embryonic stem cells.
DNA Res, 16(1):45–58, Feb 2009.

B. A. Shapiro, Y. G. Yingling, W. Kasprzak, and E. Bindewald.
Bridging the gap in rna structure prediction.
Curr Opin Struct Biol, 17(2):157–165, Apr 2007.