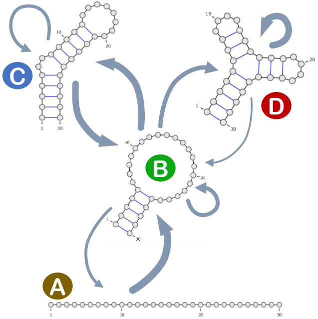
M2 BIM – STRUCT - Lecture 2

Boltzmann equilibrium

Yann Ponty

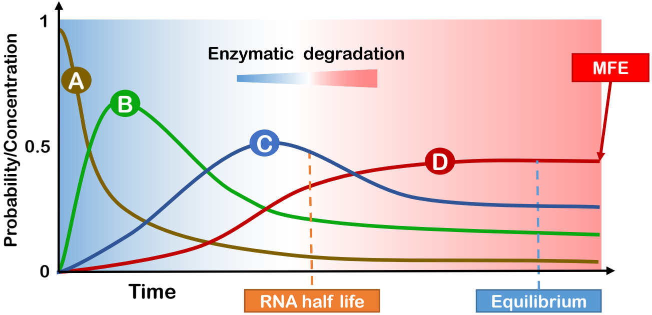
AMIBio Team
École Polytechnique/CNRS

Paradigms in RNA structural bioinformatics



A – Kinetic Landscape

Continuous-time Markov chain



B – Evolution of concentrations

Given free-energy $E : \{A, C, G, U\}^\star \times \mathcal{S} \to \mathbb{R}$, at the Boltzmann equilibrium:

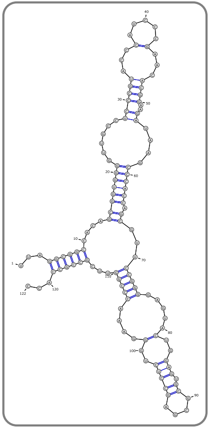
$$\mathbb{P}(S \mid w) \propto e^{-E(w,S)/RT}$$

- Minimum Free-Energy (MFE): Relevant structure = Most stable/probable
- Partition function: Equilibrium properties of Boltzmann ensemble
- Kinetics: Finite-time evolution of concentrations/probabilities

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 ...

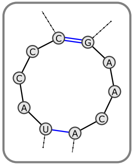
Experimentally determined
+ Interpolated for larger loops.

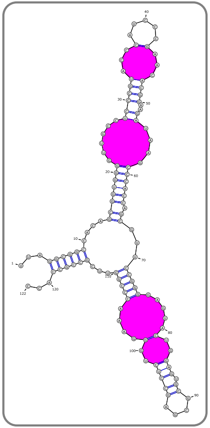
Improved results by taking stacking into account.

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 . . .

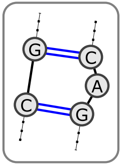
Experimentally determined
+ Interpolated for larger loops.

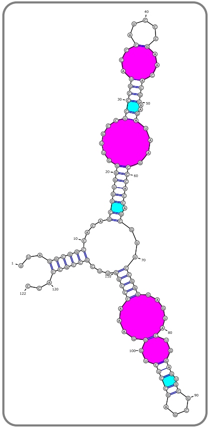
Improved results by taking stacking into account.

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 . . .

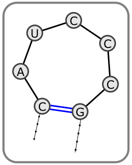
Experimentally determined
+ Interpolated for larger loops.

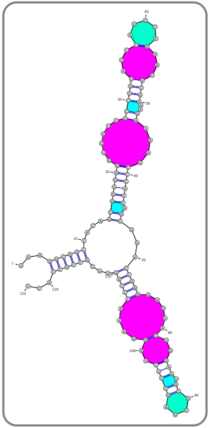
Improved results by taking stacking into account.

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 . . .

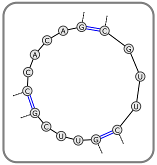
Experimentally determined
+ Interpolated for larger loops.

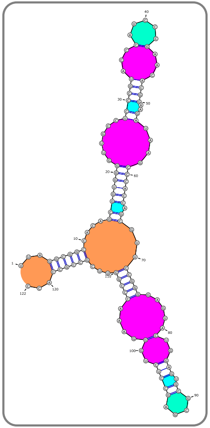
Improved results by taking stacking into account.

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 . . .

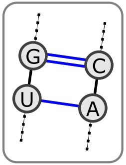
Experimentally determined
+ Interpolated for larger loops.

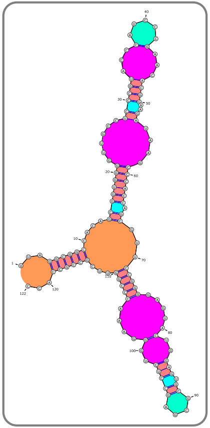
Improved results by taking stacking into account.

Turner energy model

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

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



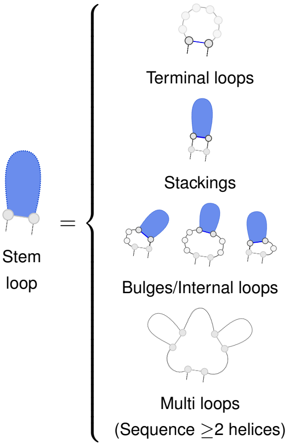


Free-energy Δ G of a loop depend on bases, assymmetry, dangles . . .

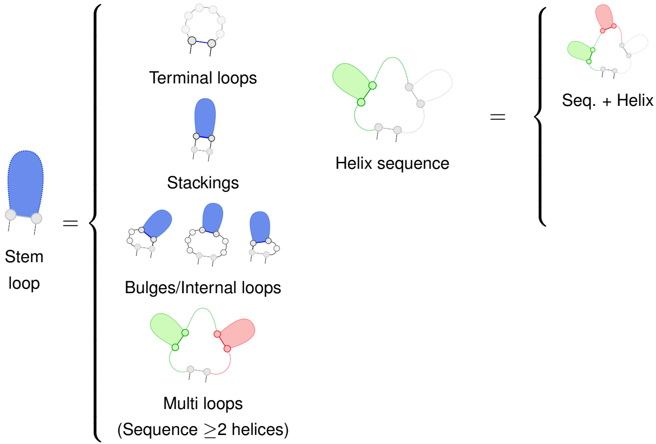
Experimentally determined
+ Interpolated for larger loops.

Improved results by taking stacking into account.

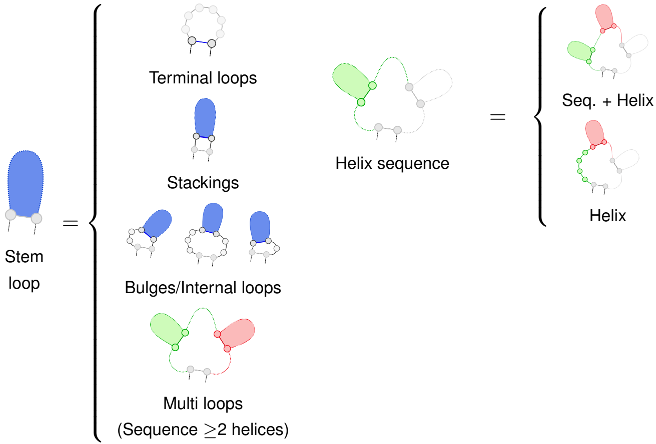
MFE DP equations



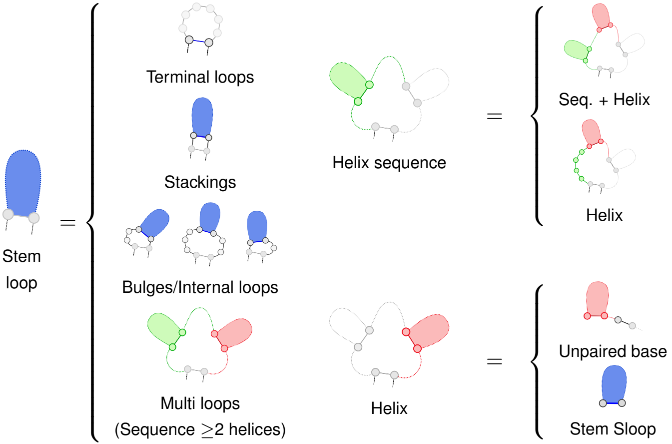
MFE DP equations



MFE DP equations

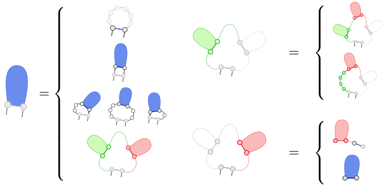


MFE DP equations



MFold Unafold

- $E_H(i,j)$: 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

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

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

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

Backtracking

Backtracking to reconstruct MFE structure:

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

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

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

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[1] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

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

Backtracking

Backtracking to reconstruct MFE structure:

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

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

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

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[1] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

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

Backtracking

Backtracking to reconstruct MFE structure:

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

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

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

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[1] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

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

Backtracking

Backtracking to reconstruct MFE structure:

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

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

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

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[1] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

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

Backtracking

Backtracking to reconstruct MFE structure:

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

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

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

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[1] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

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

Backtracking

Backtracking to reconstruct MFE structure:

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

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[1] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

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

Backtracking

Backtracking to reconstruct MFE structure:

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

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

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

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[1] time/space complexities in $\mathcal{O}(n^3)/\mathcal{O}(n^2)$

---

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

Outline

Turner energy model
MFold/Unafold

Boltzmann ensemble

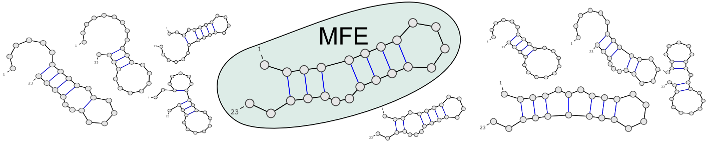
Nussinov: Minimisation $\Rightarrow$ Counting
Computing the partition function
Statistical sampling

The canonical Boltzmann Ensemble

RNA *breathes* $\Rightarrow$ 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.
$\Rightarrow$ To understand how RNA acts, one must account for the set of alternative structures.

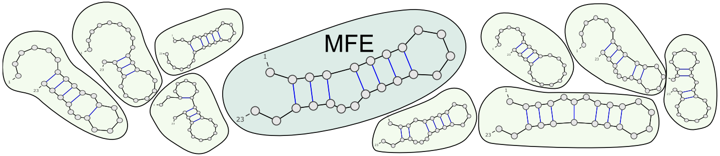
In particular, structurally close structures may *ally*, and become the most realistic candidate in the search for a functional conformation.

The canonical Boltzmann Ensemble

RNA *breathes* $\Rightarrow$ 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.
$\Rightarrow$ 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.

Boltzmann Distribution: Definition

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

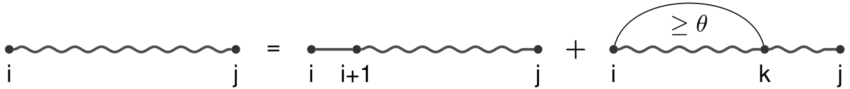
$$\mathcal{Z}_\omega = \sum_{S \in \mathcal{S}_\omega} e^{\frac{-E_{S,\omega}}{RT}}$$

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

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

Nussinov/Jacobson DP scheme

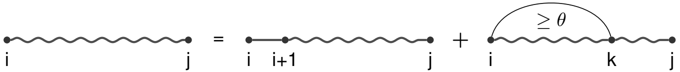


$$
\begin{aligned}
N_{i,t} &= 0, \quad \forall t \in [i, i+\theta] \\
N_{i,j} &= \min \begin{cases} N_{i+1,j} & i \text{ unpaired} \\ \min\limits_{k=i+\theta+1}^{j} \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j} & i \text{ paired with } k \end{cases}
\end{aligned}
$$

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

Nussinov/Jacobson DP scheme



Recurrence for minimal free-energy of a fold :

$$
\begin{aligned}
N_{i,t} &= 0, \quad \forall t \in [i, i+\theta] \\
N_{i,j} &= \min \begin{cases} N_{i+1,j} & (i \text{ unpaired}) \\ \min_{k=i+\theta+1}^{j} E_{i,k} + N_{i+1,k-1} + N_{k+1,j} & (i \text{ comp. with } k) \end{cases}
\end{aligned}
$$

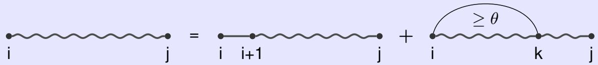
Recurrence for counting compatible structures :

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

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

Partition function

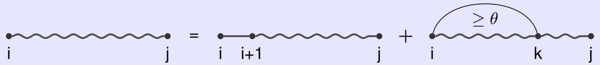
Partition function = Weighted count over compatible structures



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

Partition function

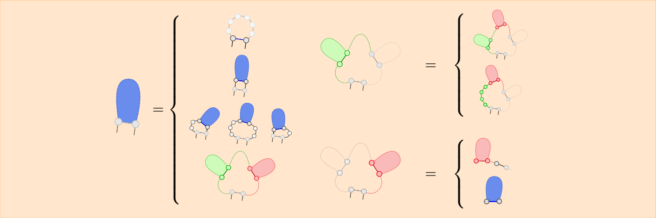
Partition function = Weighted count over compatible structures



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

Partition function

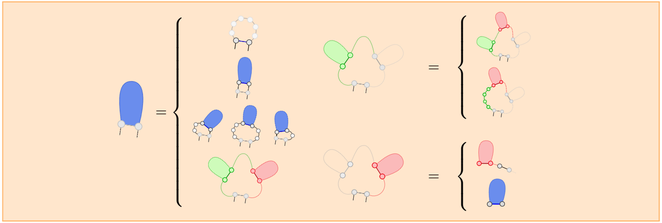
Partition function = Weighted count over compatible structures



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

Partition function

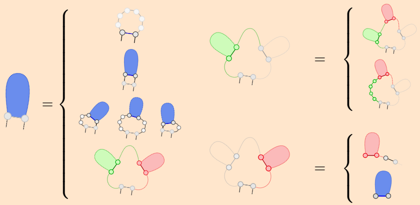
Partition function = Weighted count over compatible structures



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

Partition function

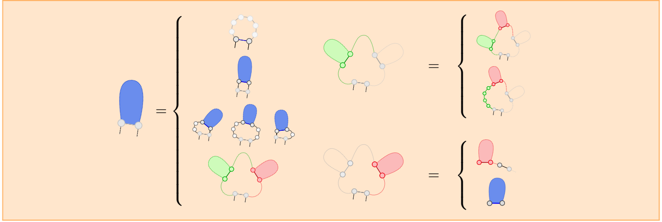
Partition function = Weighted count over compatible structures



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

Partition function

Partition function = Weighted count over compatible structures



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

Partition function

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
  $e^{-E/RT} \rightarrow$ 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

Partition function = Weighted count over compatible structures

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

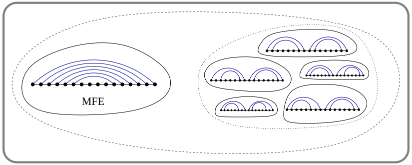
Validity of a partition function computation:

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

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

Statistical sampling of RNA 2^{ary} structures

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?

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

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

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

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

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

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

$$A_1 - A_2 - B_i - B_{i+1} - \ldots - B_{j-1} - B_j - \overset{\boxed{r}}{\overset{\downarrow}{C_i}} - C_{i+1} - \ldots - C_{j-1} - C_j$$

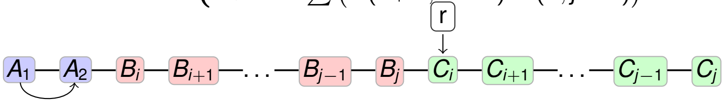
Stochastic backtrack (adapted from `SFold`)

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

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



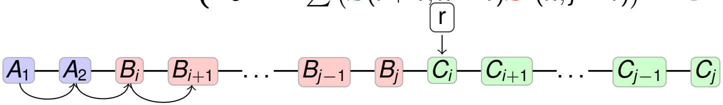
Stochastic backtrack (adapted from `SFold`)

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

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



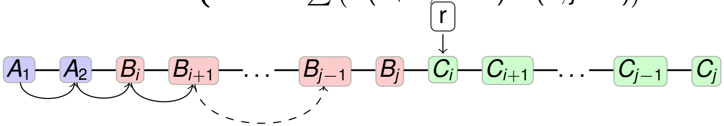
Stochastic backtrack (adapted from `SFold`)

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

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



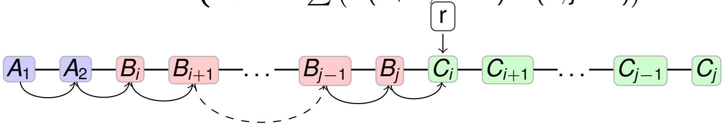
Stochastic backtrack (adapted from `SFold`)

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

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



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

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

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

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

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

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

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

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

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

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

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

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

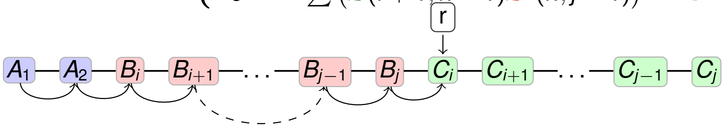
Complexity

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

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



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].

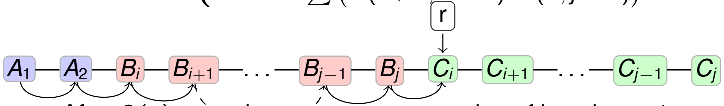
Complexity

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

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



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].

References I

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