M2 BIM – STRUCT - Lecture 2

Boltzmann equilibrium and beyond

Yann Ponty

AMIBio Team
École Polytechnique/CNRS

Outline

Physics-based structure prediction
- Turner energy model
- MFold/Unafold

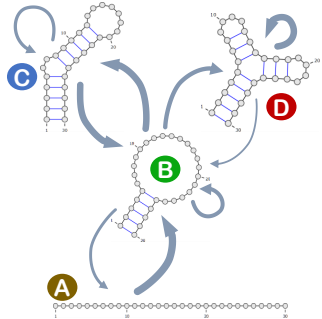
Boltzmann ensemble
- Nussinov: Minimisation $\Rightarrow$ Counting
- Computing the partition function
- Statistical sampling

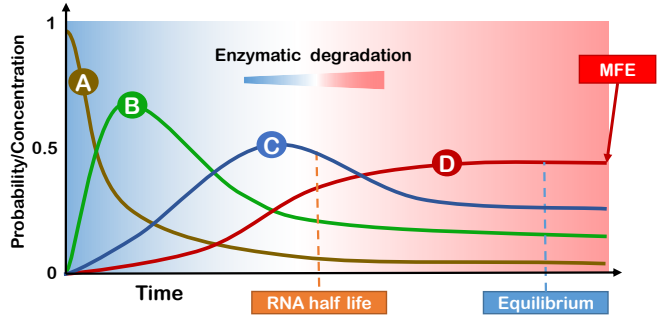
Performances
- Overall picture
- Family-level evaluation
- Evaluation issues

The specific case and issues of ML
- Deep learning: Beauty and the beast
- ML performances as advertised by authors
- Surprising limitations
- Takeaways

Extended Alorithmics/DP techniques
- Suboptimal structures
- Pseudoknots

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:

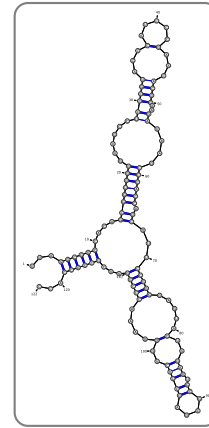
$$\mathbb{P}(S \mid w) \propto e^{-E(w,S)/RT}$$

- 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 ...

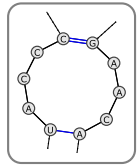
Experimentally determined
+ Interpolated for larger loops.

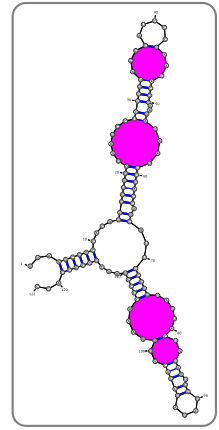
Improved results by taking stacking into account.

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 ...

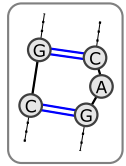
Experimentally determined
+ Interpolated for larger loops.

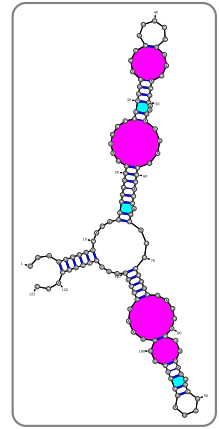
Improved results by taking stacking into account.

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 ...

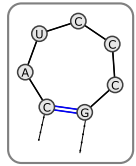
Experimentally determined
+ Interpolated for larger loops.

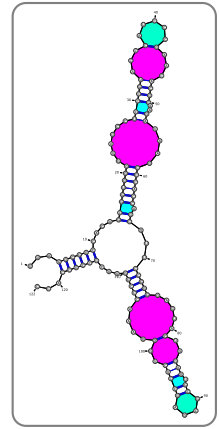
Improved results by taking stacking into account.

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 ...

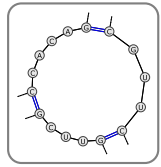
Experimentally determined
+ Interpolated for larger loops.

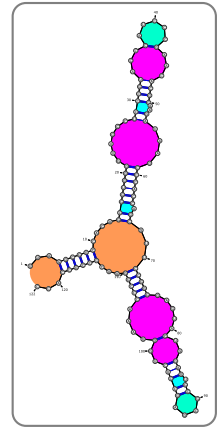
Improved results by taking stacking into account.

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 ...

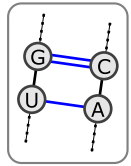
Experimentally determined
+ Interpolated for larger loops.

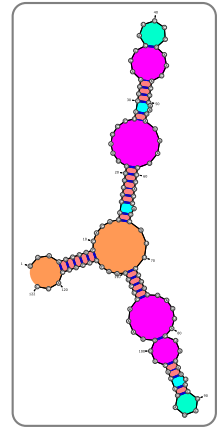
Improved results by taking stacking into account.

Turner energy model

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

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



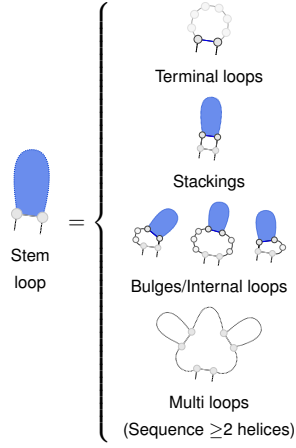


Free-energy Δ G of a loop depend on bases, assymmetry, dangles ...

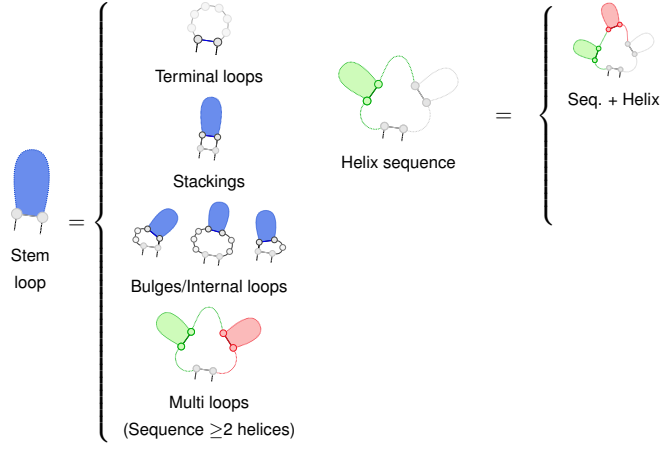
Experimentally determined
+ Interpolated for larger loops.

Improved results by taking stacking into account.

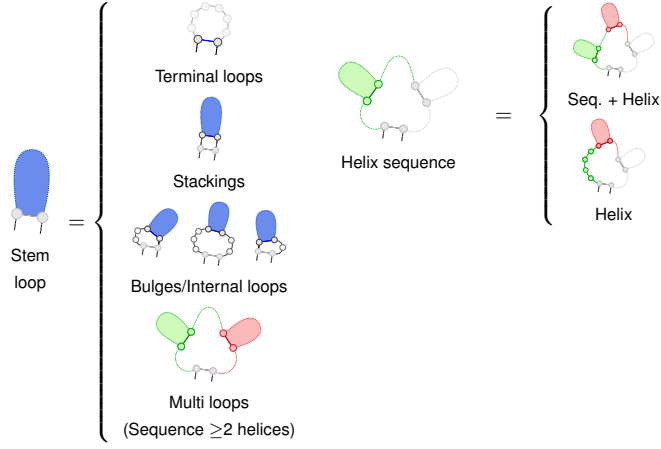
MFE DP equations



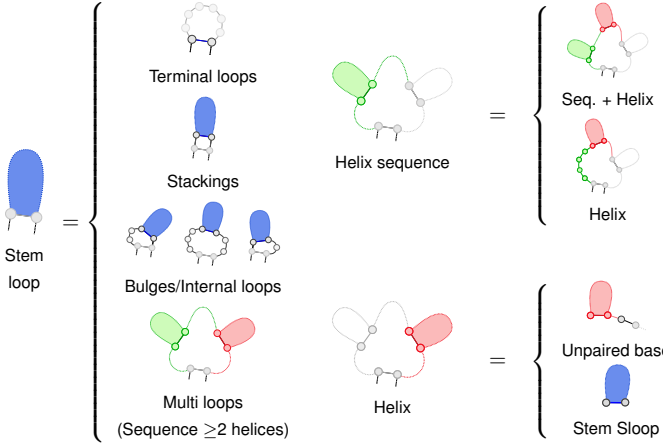
MFE DP equations



MFE DP equations

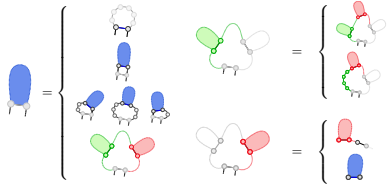


MFE DP equations



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

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

Backtracking

Backtracking to reconstruct MFE structure:

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

Complexity:

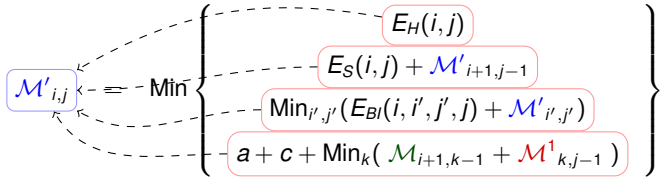
For each min, $\mathcal{O}(n)$ potential contributors
$\Rightarrow$ Worst-case complexity in $\mathcal{O}(n^2)$ for naive backtrack.
Keep best contributor for each Min $\Rightarrow$ Backtracking in $\mathcal{O}(n)$

$\Rightarrow$ `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:



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

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

Complexity:

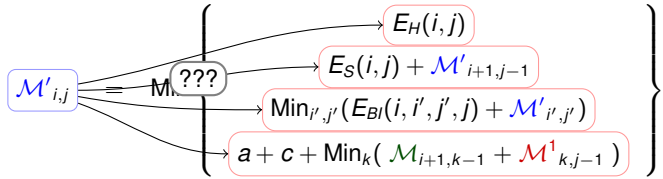
For each min, $\mathcal{O}(n)$ potential contributors
$\Rightarrow$ Worst-case complexity in $\mathcal{O}(n^2)$ for naive backtrack.
Keep best contributor for each Min $\Rightarrow$ Backtracking in $\mathcal{O}(n)$

$\Rightarrow$ `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:



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

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

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:

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

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

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

Complexity:

For each min, $\mathcal{O}(n)$ potential contributors
$\Rightarrow$ 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:

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

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

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

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:

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

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:

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

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...

Outline

Physics-based structure prediction
- Turner energy model
- MFold/Unafold

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

Performances
- Overall picture
- Family-level evaluation
- Evaluation issues

The specific case and issues of ML
- Deep learning: Beauty and the beast
- ML performances as advertised by authors
- Surprising limitations
- Takeaways

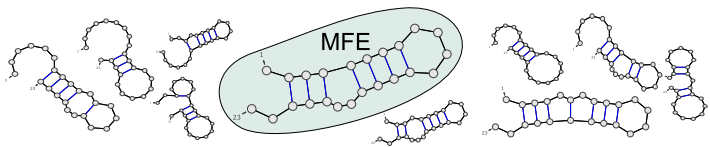
Extended Alorithmics/DP techniques
- Suboptimal structures
- Pseudoknots

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.

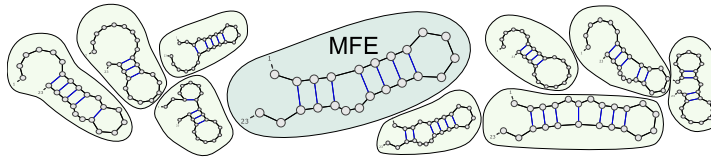
In particular, structurally close structures may *ally*, and become the most realistic candidate in the search for a functional conformation.

The canonical Boltzmann Ensemble

RNA *breathes* ⇒ 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.

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

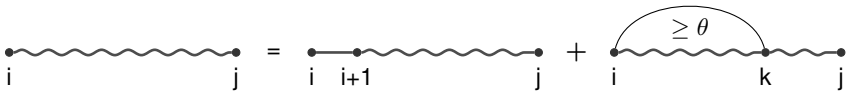
$$\mathcal{Z}_\omega = \sum_{S \in \mathcal{S}_\omega} e^{\frac{-E_{S,\omega}}{RT}}$$

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

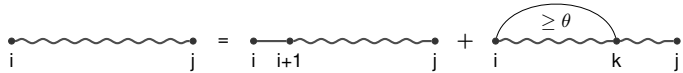


$$
\begin{aligned}
N_{i,t} &= 0, \quad \forall t \in [i, i+\theta] \\
N_{i,j} &= \min \begin{cases} N_{i+1,j} & i \text{ unpaired} \\ \min\limits_{k=i+\theta+1}^{j} \Delta G_{i,k} + N_{i+1,k-1} + N_{k+1,j} & i \text{ paired with } k \end{cases}
\end{aligned}
$$

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 :

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

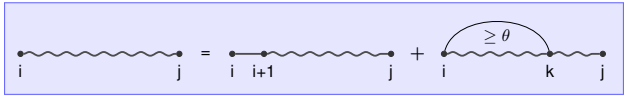
Recurrence for counting compatible structures :

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

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

Partition function

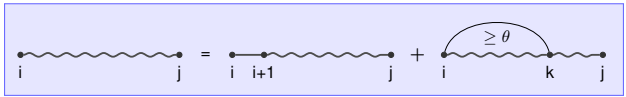
Partition function = Weighted count over compatible structures



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

Partition function

Partition function = Weighted count over compatible structures

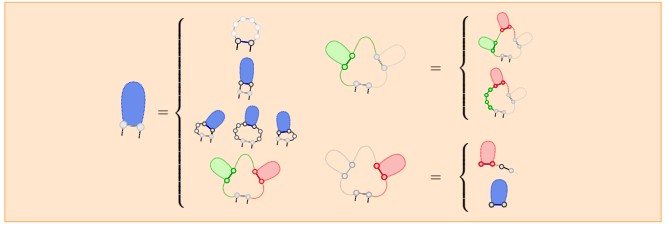


$$\mathcal{Z}_{i,t} = 1, \quad \forall t \in [i, i+\theta]$$

$$\mathcal{Z}_{i,j} = \sum \left\{ \sum_{k=i+\theta+1}^{j} e^{\frac{-E_{bp}(i,k)}{RT}} \times \mathcal{Z}_{i+1,k-1} \times \mathcal{Z}_{k+1,j} \right. \mathcal{Z}_{i+1,j}$$

Partition function

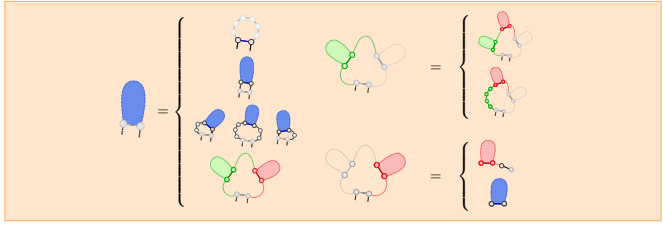
Partition function = Weighted count over compatible structures



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

Partition function

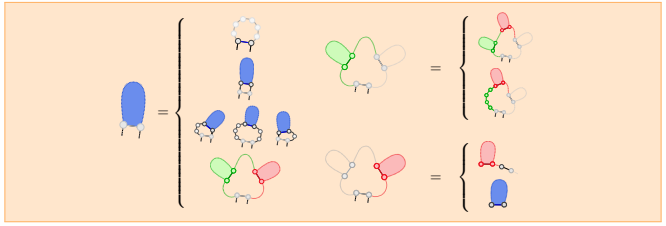
Partition function = Weighted count over compatible structures



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

Partition function

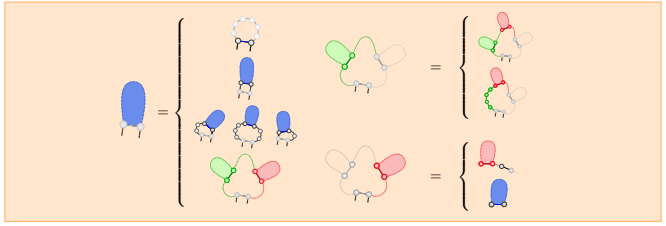
Partition function = Weighted count over compatible structures



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

Partition function

Partition function = Weighted count over compatible structures



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

Partition function

Partition function = Weighted count over compatible structures

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

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

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

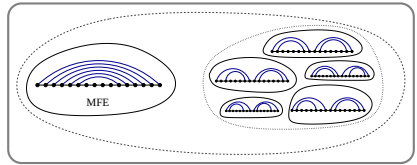
Validity of a partition function computation:

- 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$^{\text{ary}}$ 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

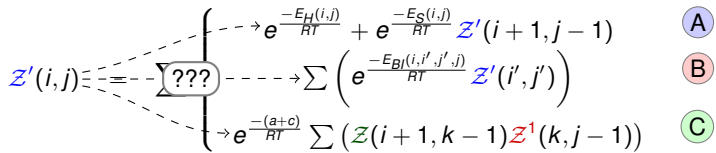
How to sample from the Boltzmann ensemble?

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



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

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

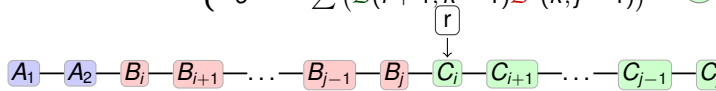
Stochastic backtrack (adapted from `SFold`)

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

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



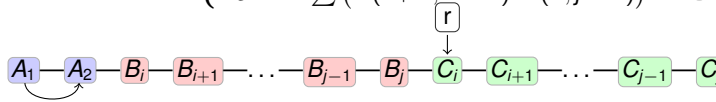
Stochastic backtrack (adapted from `SFold`)

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

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



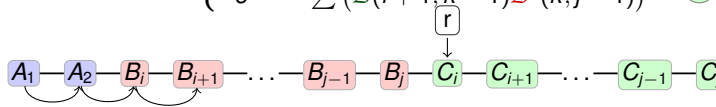
Stochastic backtrack (adapted from `SFold`)

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

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



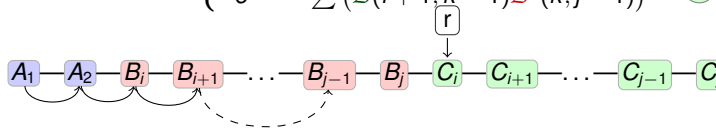
Stochastic backtrack (adapted from `SFold`)

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

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



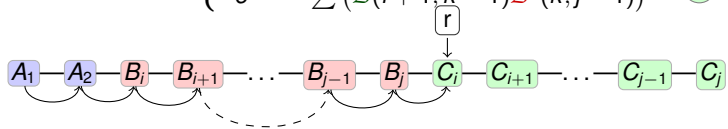
Stochastic backtrack (adapted from `SFold`)

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

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



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

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

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

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

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

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

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

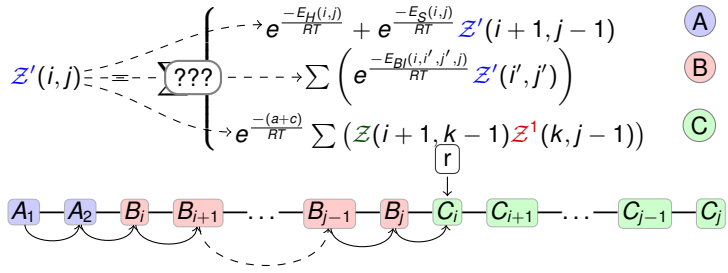
$$p_S = \frac{\mathcal{B}(\{S\})}{\mathcal{B}(\mathcal{S}_w)} = \frac{e^{-E_S/RT}}{\mathcal{Z}} = P_{S,\omega}$$

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



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].

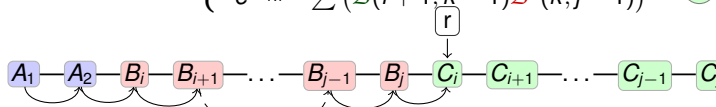
Complexity

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].

Outline

- Physics-based structure prediction
  - Turner energy model
  - MFold/Unafold
- Boltzmann ensemble
  - Nussinov: Minimisation $\Rightarrow$ Counting
  - Computing the partition function
  - Statistical sampling
- Performances
  - Overall picture
  - Family-level evaluation
  - Evaluation issues
- The specific case and issues of ML
  - Deep learning: Beauty and the beast
  - ML performances as advertised by authors
  - Surprising limitations
  - Takeaways
- Extended Alorithmics/DP techniques
  - Suboptimal structures
  - Pseudoknots

Historical paradigms towards 2D prediction

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

Historical paradigms towards 2D prediction

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.

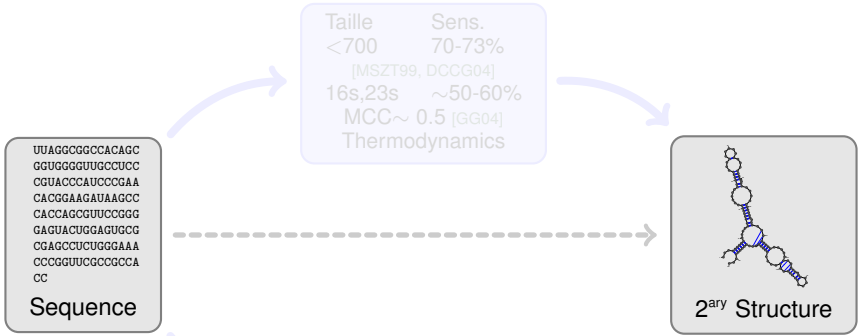
Avantages :

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

Limitations

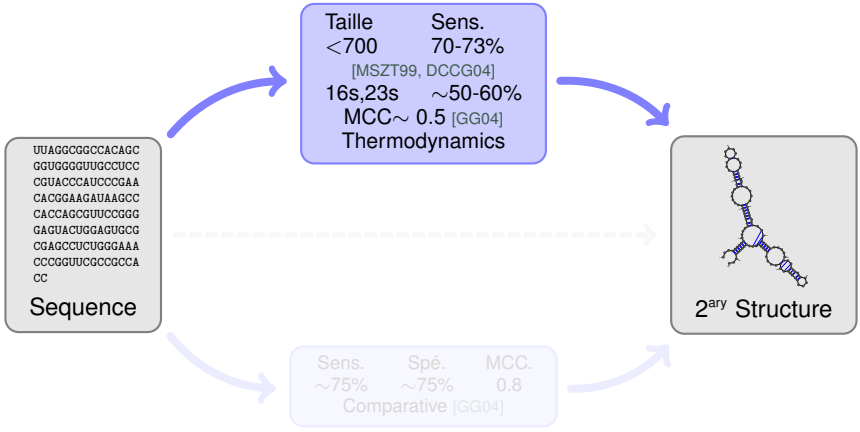
- Easily unreasonable complexity
- Non exhaustive search
- Captures *transient* structures

Typical performances



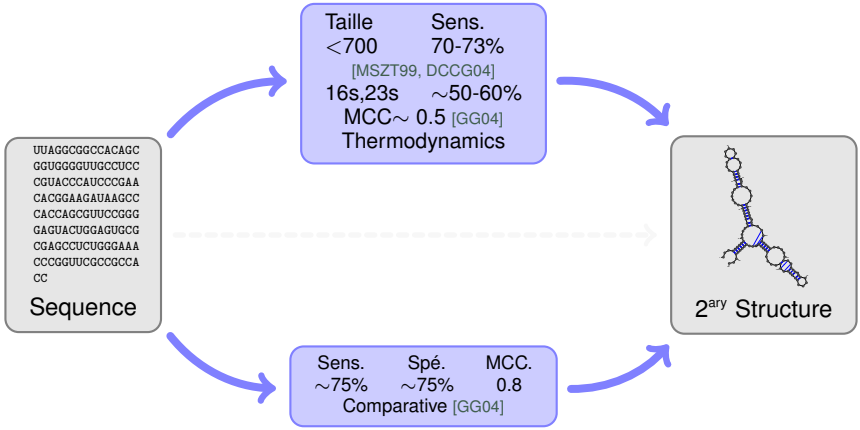
Reminder: $MCC = \frac{t^+t^- - f^+f^-}{\sqrt{(t^+ + f^+)(t^+ + f^-)(t^- + f^+)(t^- + f^-)}}$

Typical performances



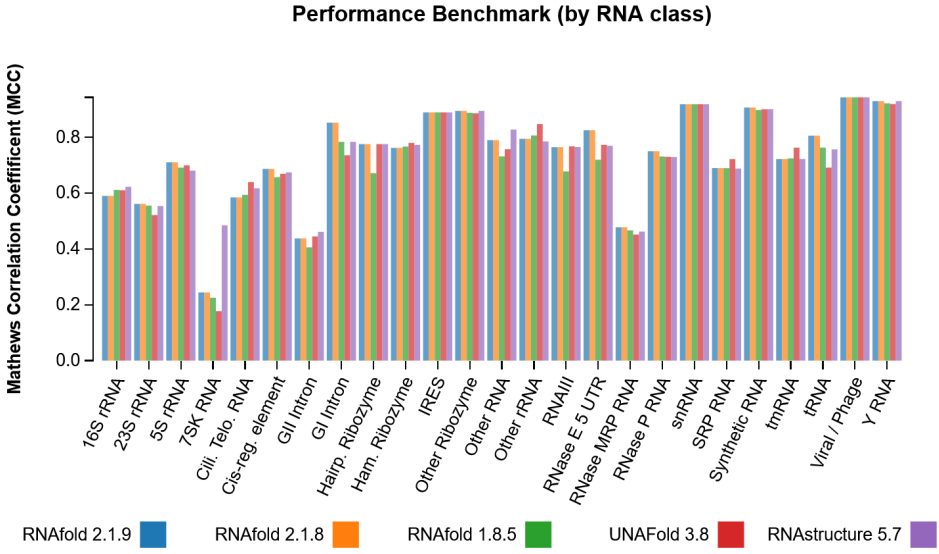
Reminder: $MCC = \frac{t^+t^- - f^+f^-}{\sqrt{(t^+ + f^+)(t^+ + f^-)(t^- + f^+)(t^- + f^-)}}$

Typical performances



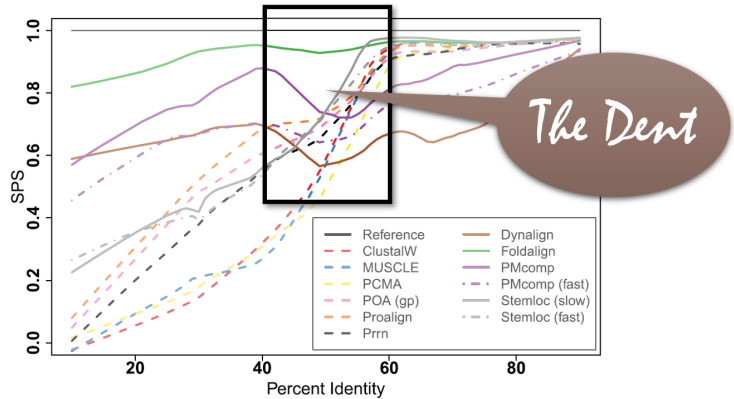
Reminder: $MCC = \frac{t^+t^- - f^+f^-}{\sqrt{(t^+ + f^+)(t^+ + f^-)(t^- + f^+)(t^- + f^-)}}$

Detailed performances of 2D folding algorithms



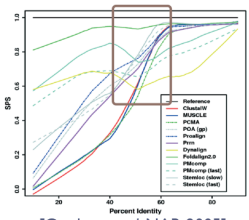
Biased benchmarks: precedent in comparative folding/alignment

Bralibase: Benchmark for comp. RNA folding [Gardner,Wilm & Washietl, NAR 2005]



[Löwes *et al*, Briefings in Bioinfo 2016]

Biased benchmarks: precedent in comparative folding/alignment



[Gardner *et al*, NAR 2005]

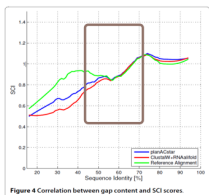
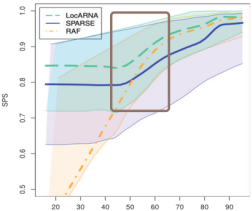
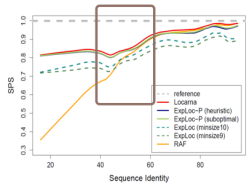


Figure 4 Correlation between gap content and SCI scores.

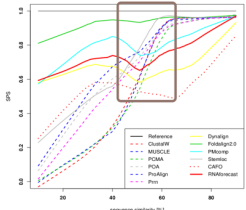
[Bremges *et al*, BMC Bioinfo, 2010]



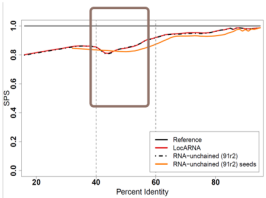
[Will *et al*, Bioinformatics 2015]



[Schmiedl *et al*, RECOMB 2012]



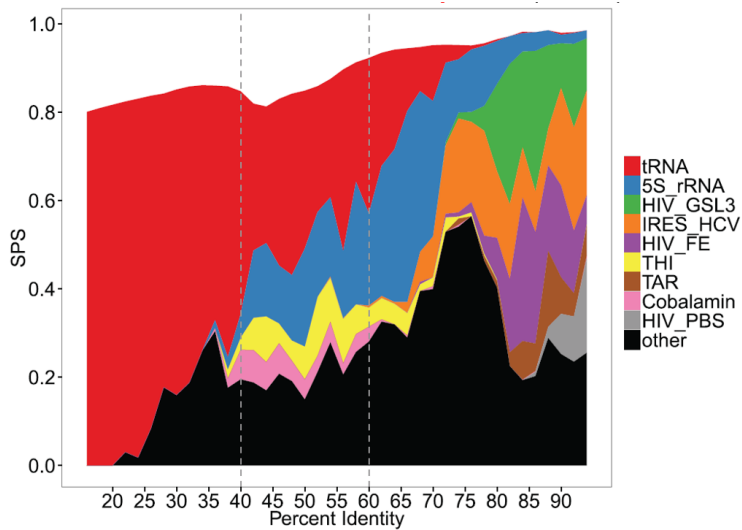
[Höchsmann *et al*, Unpublished]



[Bourgeade *et al*, J Comp Biol, 2015]

[Löwes *et al*, Briefings in Bioinfo 2016]

Biased benchmarks: precedent in comparative folding/alignment



[Löwes *et al*, Briefings in Bioinfo 2016]

Outline

Physics-based structure prediction
- Turner energy model
- MFold/Unafold

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

Performances
- Overall picture
- Family-level evaluation
- Evaluation issues

The specific case and issues of ML
- Deep learning: Beauty and the beast
- ML performances as advertised by authors
- Surprising limitations
- Takeaways

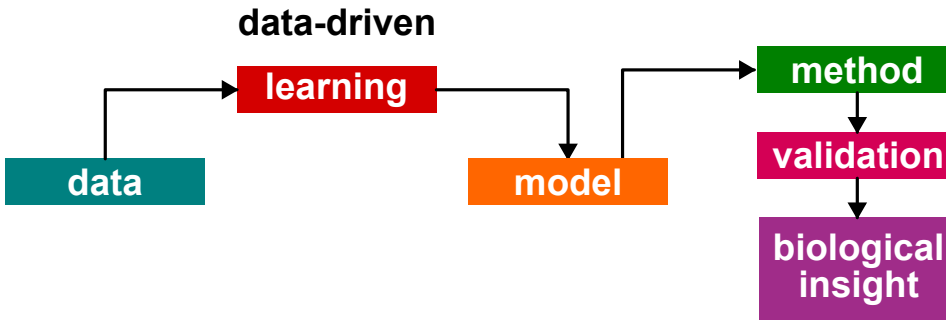
Extended Alorithmics/DP techniques
- Suboptimal structures
- Pseudoknots

The elephant in the room – **2010s version**

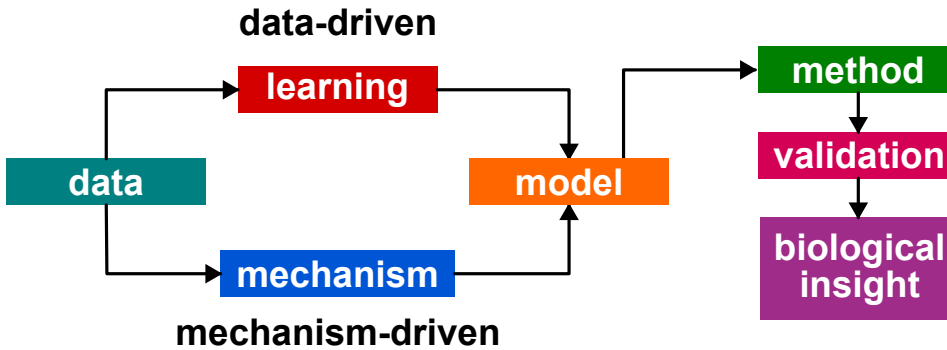
The elephant in the room – **2020s version**

A personal take on predictive Bioinformatics

A personal take on predictive Bioinformatics



A personal take on predictive Bioinformatics



Method dev. as a modeling discipline:

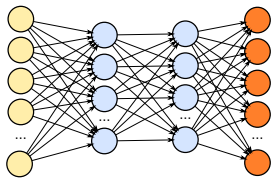
Mechanism-driven model + Exact/deterministic algorithms

$\rightarrow$ Performance as (in)validation of model

Machine Learning (ML): The beauty...

Machine Learning as a tool for scientific discovery

- Great promises
- Self-improving methods
- Generates/prioritizes hypotheses
- Available workforce (ubiquitous in curriculums)
- Highly promoted/funded by research institutions and glamorous journals...



Machine Learning (ML): The beauty... and the beast

Multiple (potential) pitfalls for ML in Bio*:

- Tricky evaluation (data leakage) $\rightarrow$ Extrapolation/generalization???
- Reproducibility issues (code/datasets availability, stability, retraining)
- Fishing expeditions/storytelling, selective reporting
- Educational deadend?
- Future(?) ecological disaster? Random blue checkmarks AI zealots on Twitter (grumble...)

(a) Husky classified as wolf (b) Explanation

[Ribeiro et al, KDD'16]

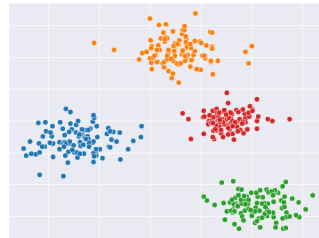
Machine Learning (ML): The beauty... and the beast

Multiple (potential) pitfalls for ML in Bio*:

- Tricky evaluation (data leakage) $\rightarrow$ Extrapolation/generalization???
- Reproducibility issues (code/datasets availability, stability, retraining)
- Fishing expeditions/storytelling, selective reporting
- Educational deadend?
- Future(?) ecological disaster? Random blue checkmarks AI zealots on Twitter (grumble...)

(a) Husky classified as wolf (b) Explanation

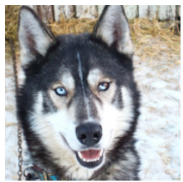
[Ribeiro et al, KDD'16]



Machine Learning (ML): The beauty... and the beast

Multiple (potential) pitfalls for ML in Bio*:

- Tricky evaluation (data leakage) $\rightarrow$ Extrapolation/generalization???
- Reproducibility issues (code/datasets availability, stability, retraining)
- Fishing expeditions/storytelling, selective reporting
- Educational deadend?
- Future(?) ecological disaster? Random blue checkmarks AI zealots on Twitter (grumble...)



(a) Husky classified as wolf

(b) Explanation

[Ribeiro et al, KDD'16]

Machine Learning (ML): The beauty... and the beast

Multiple (potential) pitfalls for ML in Bio*:

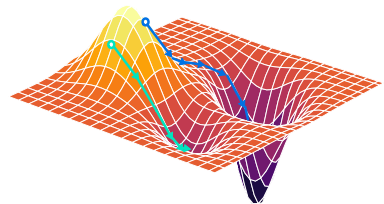
- Tricky evaluation (data leakage) → Extrapolation/generalization???
- Reproducibility issues (code/datasets availability, stability, retraining)
- Fishing expeditions/storytelling, selective reporting
- Educational deadend?
- Future(?) ecological disaster? Random blue checkmarks AI zealots on Twitter (grumble...)

Available upon request

***aka* iff I'm in a good mood,**

PhD/postdoc still in lab, HDDs haven't burned,

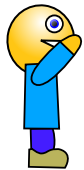
pharma hasn't expressed interest in data...



Machine Learning (ML): The beauty... and the beast

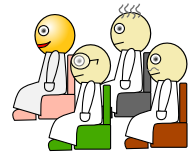
Multiple (potential) pitfalls for ML in Bio*:

- Tricky evaluation (data leakage) → Extrapolation/generalization???
- Reproducibility issues (code/datasets availability, stability, retraining)
- Fishing expeditions/storytelling, selective reporting
- Educational deadend?
- Future(?) ecological disaster? Random blue checkmarks AI zealots on Twitter (grumble...)



Fifth law of thermodynamics (continued)

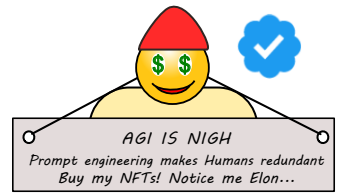
```
                    ...
-0.31622776601683794 0.31622776601683794
-0.3157663248679193  0.3160839282916222
0.006806069733149146 0.17777128902976705
0.4472135954999579   1.433348584081719
-1.5736761136523203  1.433348584081719
-0.0002340648727882  0.4522609460629265
                    ...
                              24235/1020400
```



Machine Learning (ML): The beauty... and the beast

Multiple (potential) pitfalls for ML in Bio*:

- Tricky evaluation (data leakage) → Extrapolation/generalization???
- Reproducibility issues (code/datasets availability, stability, retraining)
- Fishing expeditions/storytelling, selective reporting
- Educational deadend?
- Future(?) ecological disaster? Random blue checkmarks AI zealots on Twitter (grumble...)



A crowded ML field for RNA 2D prediction

Method	Output	PKs?	Architecture	Availability
CONTRAfold	Pairwise contacts	No	CLLM	Code+weights+webserver
EternaFold	Pairwise contacts	No	CLLM	Code+weights+webserver
DMfold	DBN	Yes	bi-LSTM	Code only
RNA-state-inf	Binary paired/unpaired	N/A	bi-LSTM	Code only
SPOT-RNA2	Pairwise contacts	Yes	CNN	Code+weights+webserver
CROSS	Binary paired/unpaired	N/A	CNN-like	Webserver
RPRes	Binary paired/unpaired	N/A	bi-LSTM+CNN	Code only
2dRNA	Pairwise contacts	Yes	bi-LSTM+CNN	Webserver
2dRNA-LD	Pairwise contacts	Yes	bi-LSTM+CNN	Webserver
SPOT-RNA	Pairwise contacts	Yes	CNN+bi-LSTM	Code+weights+webserver
MXfold2	Pseudo-dG	No	CNN+bi-LSTM	Code+weights+webserver
CNNFold	Pairwise contacts	Yes	CNN(NxN input)	Code+weights
UFold	Pairwise contacts	Yes	CNN(NxN input)	Code+weights+webserver
CDPfold	DBN	No	CNN(N×Ninput)	Code
E2Efold	Pairwise contacts	Yes	Transformer+CNN	Code+weights
ATTfold	Pairwise contacts	Yes	Transformer+CNN	No

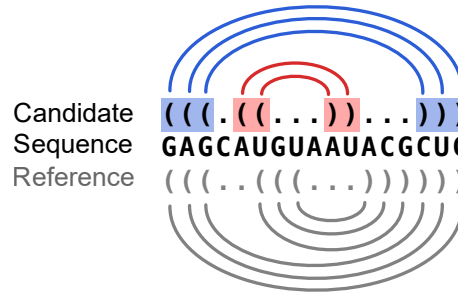
[Wu *et al*, Briefings in Bioinfo 2023]

Performances of 2D structure prediction

RNAStrand benchmark

[Adronescu *et al*, BMC Bioinf 2008]

Method	F_1
RNAfold 1.8.5	0.737
UNAfold 3.8	0.725
RNAstructure 5.7	0.744

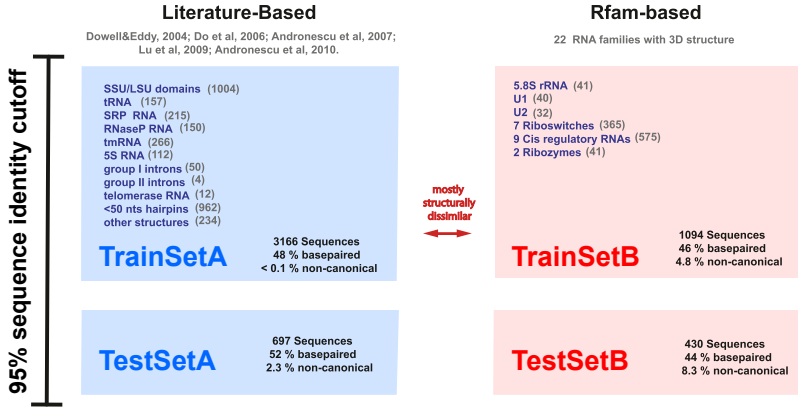


Sens = 3/6 = 0.5 PPV = 3/5 = 0.6

$F_1 = 0.545...$

$$F_1\text{-score} = \frac{2 \times \text{PPV} \times \text{Sens}}{\text{PPV} + \text{Sens}}$$

The TORNADO dataset



[Rivas *et al*, RNA 2012]

TrainSetA vs TestSetA: 95% sim. cutoff → Learn *k*-mer to template association

(May happen even for extreme cutoffs)

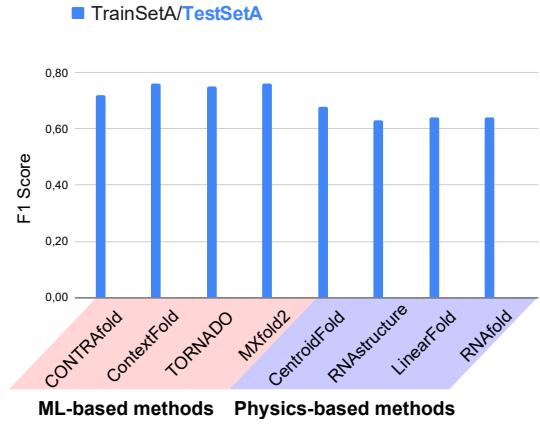
TrainSetA vs TestSetB: Rewards learning structurally generalizable models

Performances of 2D structure prediction

RNAStrand benchmark

[Adronescu *et al*, BMC Bioinf 2008]

Method	F_1
RNAfold 1.8.5	0.737
UNAfold 3.8	0.725
RNAstructure 5.7	0.744



[Sato *et al*, Nature Comm 2021]

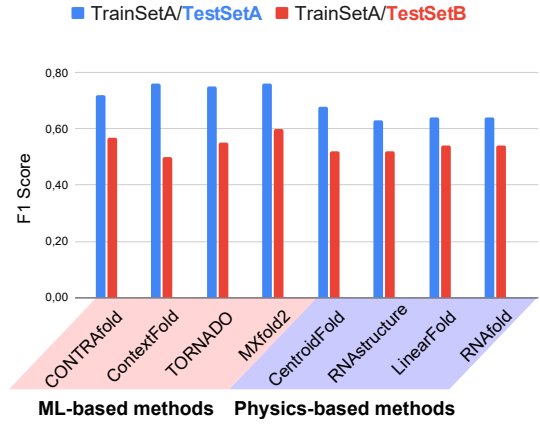
$$F_1\text{-score} = \frac{2 \times \text{PPV} \times \text{Sens}}{\text{PPV} + \text{Sens}}$$

Performances of 2D structure prediction

RNAStrand benchmark

[Adronescu *et al*, BMC Bioinf 2008]

Method	F_1
RNAfold 1.8.5	0.737
UNAfold 3.8	0.725
RNAstructure 5.7	0.744

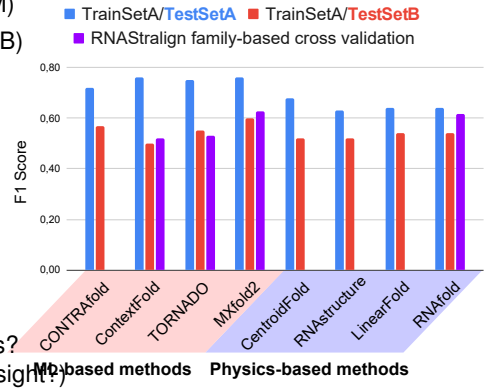


[Sato *et al*, Nature Comm 2021]

$$F_1\text{-score} = \frac{2 \times \text{PPV} \times \text{Sens}}{\text{PPV} + \text{Sens}}$$

The (nc)RNA datasphere

- 34M sequences, inc 22M presumably structured (RNACentral)
- 4000+ functional ncRNA families (RFAM)
- 250-300 non-redundant 3D models (PDB)



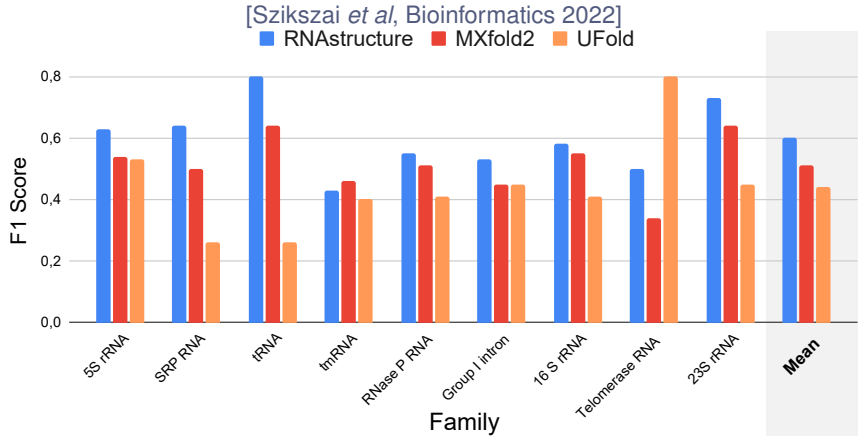
Existing methods trained on datasets:

- highly-redundant sequence-wise
- low-diversity structure-wise

Do ML methods generalize to new structures?
(Do ML perfs translate into *new* biological insight?)

[Sato *et al*, Nature Comm 2021]

Generalization to new families/structures remains problematic

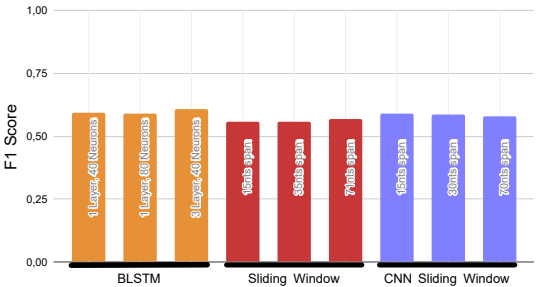


Family-fold cross-validation on ArchiveII dataset [Sloma & Mathews, RNA 2016]
3974 RNAs of length 77-438 (large rRNAs split into smaller domains)

What if you had access to (unbounded) additional data?

Idea: Assess NN models' capacity to emulate RNAfold on random sequences

[Flamm *et al*, Frontiers in Bioinfo 2022]

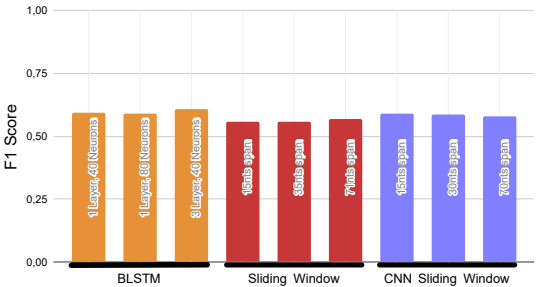


Perfs *plateau* at 80k seq/structs (70nts)

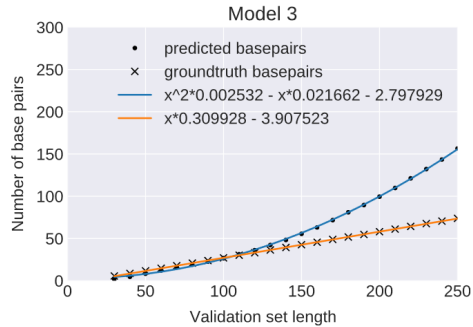
What if you had access to (unbounded) additional data?

Idea: Assess NN models' capacity to emulate RNAfold on random sequences

[Flamm *et al*, Frontiers in Bioinfo 2022]

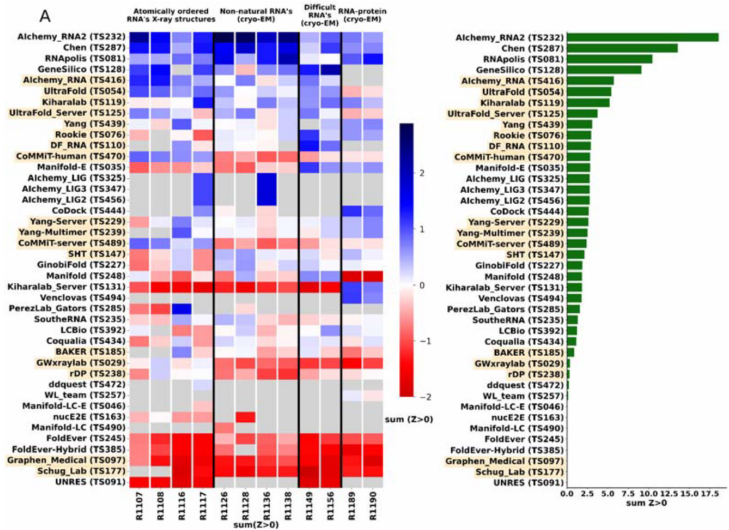


Perfs *plateau* at 80k seq/structs (70nts)



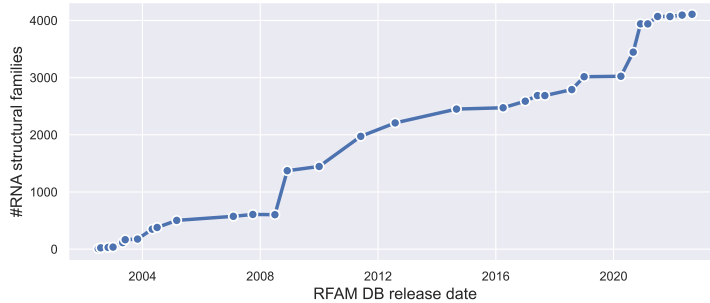
Popular CNN predicts $\Theta(n^2)$ BPs!

RNA 3D structure: No AlphaFold moment at CASP15



[Das *et al*, under review]

Conclusions and musings



- Still a need for improved RNA prediction (possibly ML-based)
- Purely combinatorial methods still $\pm$ state-of-the-art for new families...
- Hybrid approaches *à la* MxFold2: Best of both worlds?
- Assessing intrinsic limits of architectures: RNAFold as surrogate model

Conclusions and musings

So what's special about RNA?

- Modular but combinatorial structure
- New folds being routinely discovered (+ can be designed)
- Relatively scarce 3D data
- Opportunity: Tons of probing data (ML)
- Potential of LLMs/transformers (incoming)
- Pseudoknots-ready algorithms

Conclusions and musings

RNA/Bioinfo community needs to enforce stricter standards for ML publications:

- Enforce datasets and source code availability
  [Szikszai *et al*, Bioinfo'22] found 4/8 recent DL methods non-functional
- Realistic retraining mandatory
  Precondition for self-improvement, benchmarking of novel methods
- Consider mechanistic and ML methods as largely incomparable
- Better datasets/benchmarks needed, but perhaps not sufficient
- Sequence-based leakage should be systematically investigated

Outline

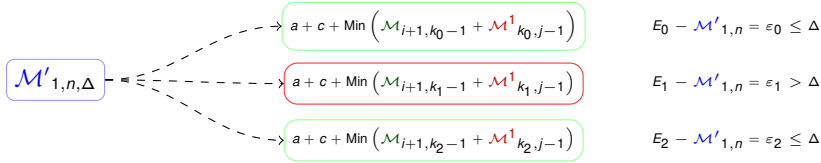
- Physics-based structure prediction
  - Turner energy model
  - MFold/Unafold
- Boltzmann ensemble
  - Nussinov: Minimisation $\Rightarrow$ Counting
  - Computing the partition function
  - Statistical sampling
- Performances
  - Overall picture
  - Family-level evaluation
  - Evaluation issues
- The specific case and issues of ML
  - Deep learning: Beauty and the beast
  - ML performances as advertised by authors
  - Surprising limitations
  - Takeaways
- Extended Alorithmics/DP techniques
  - Suboptimal structures
  - Pseudoknots

Suboptimal structures

Prob.: Simplified energy model (no pseudoknots, only canonical BPs)
$\Rightarrow$ Native structure (functional) could be overthrown.

$\Rightarrow$ Investigate suboptimal structures (`RNASubopt` [WFHS99]),
i.e. build all structures within Δ KCal.mol^{-1} of MFE:

- Compute minimum free-energy matrices
- Backtrack on any contribution within Δ of MFE;
- Update Δ such that future backtracks create ≥ 1 struct.
- Recursively generate subopts and combine (brute-force ou Sort)

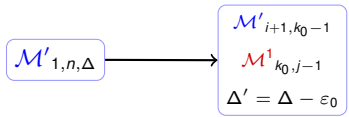


Suboptimal structures

Prob.: Simplified energy model (no pseudoknots, only canonical BPs)
$\Rightarrow$ Native structure (functional) could be overthrown.

$\Rightarrow$ Investigate suboptimal structures (`RNASubopt` [WFHS99]),
i.e. build all structures within Δ KCal.mol^{-1} of MFE:

- Compute minimum free-energy matrices
- Backtrack on any contribution within Δ of MFE;
- Update Δ such that future backtracks create ≥ 1 struct.
- Recursively generate subopts and combine (brute-force ou Sort)

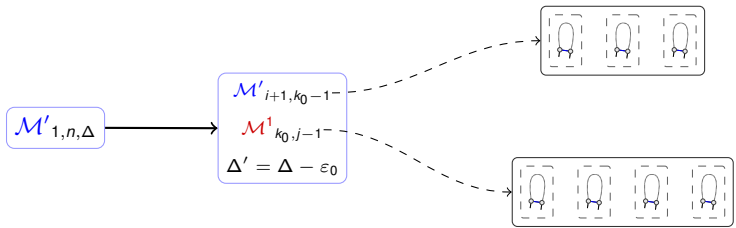


Suboptimal structures

Prob.: Simplified energy model (no pseudoknots, only canonical BPs)
$\Rightarrow$ Native structure (functional) could be overthrown.

$\Rightarrow$ Investigate suboptimal structures (`RNASubopt` [WFHS99]),
i.e. build all structures within Δ KCal.mol^{-1} of MFE:

- Compute minimum free-energy matrices
- Backtrack on any contribution within Δ of MFE;
- Update Δ such that future backtracks create ≥ 1 struct.
- Recursively generate subopts and combine (brute-force ou Sort)

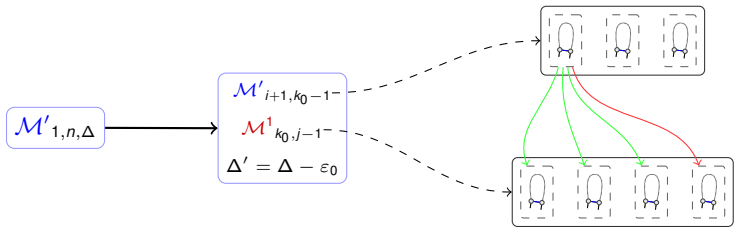


Suboptimal structures

Prob.: Simplified energy model (no pseudoknots, only canonical BPs)
$\Rightarrow$ Native structure (functional) could be overthrown.

$\Rightarrow$ Investigate suboptimal structures (`RNASubopt` [WFHS99]),
i.e. build all structures within Δ KCal.mol^{-1} of MFE:

- Compute minimum free-energy matrices
- Backtrack on any contribution within Δ of MFE;
- Update Δ such that future backtracks create ≥ 1 struct.
- Recursively generate subopts and combine (brute-force ou Sort)

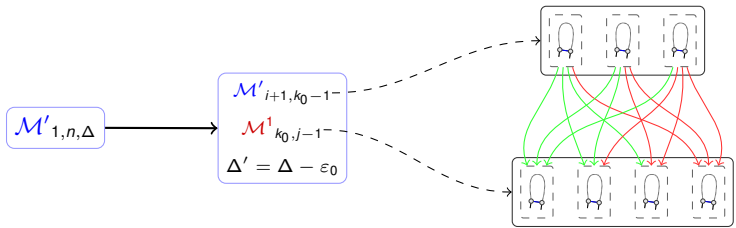


Suboptimal structures

Prob.: Simplified energy model (no pseudoknots, only canonical BPs)
$\Rightarrow$ Native structure (functional) could be overthrown.

$\Rightarrow$ Investigate suboptimal structures (`RNASubopt` [WFHS99]),
i.e. build all structures within Δ KCal.mol^{-1} of MFE:

- Compute minimum free-energy matrices
- Backtrack on any contribution within Δ of MFE;
- Update Δ such that future backtracks create ≥ 1 struct.
- Recursively generate subopts and combine (brute-force ou Sort)



Suboptimal structures

Prob.: Simplified energy model (no pseudoknots, only canonical BPs)
$\Rightarrow$ Native structure (functional) could be overthrown.

$\Rightarrow$ Investigate suboptimal structures (`RNASubopt` [WFHS99]),
i.e. build all structures within Δ KCal.mol^{-1} of MFE:

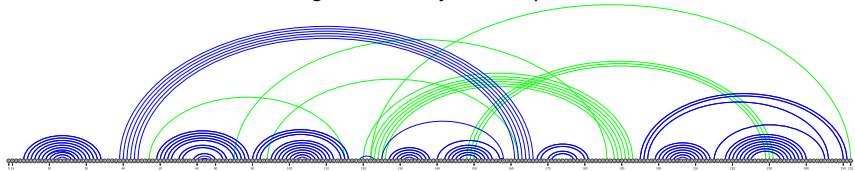
- Compute minimum free-energy matrices
- Backtrack on any contribution within Δ of MFE;
- Update Δ such that future backtracks create ≥ 1 struct.
- Recursively generate subopts and combine (brute-force ou Sort)

$$\Rightarrow \text{Time complexity (Sort)} : \mathcal{O}(n^3 + n \cdot k \log(k))$$

(k grows exponentially fast with Δ!)

Predicting pseudoknotted structures

Pseudoknots are essential to the folding and activity of multiple RNA families.



Groupe I Ribozyme

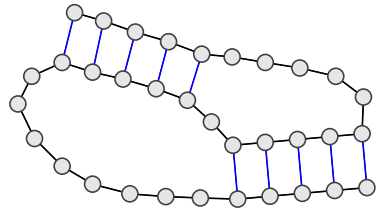
Their disregard within current folding algorithms stems both from **algorithmic** and **energetic** intricacies.

(Pseudoknots = Crossings $\Rightarrow$ foldings delimited by base-pair can no longer be assumed to be independent)

Type	Complexity	Reference
Secondary structures	$\mathcal{O}(n^3)$	[MSZT99]
L&P	$\mathcal{O}(n^5)$	[LP00]
D&P	$\mathcal{O}(n^5)$	[DP03]
A&U	$\mathcal{O}(n^5)$	[Aku00]
R&E	$\mathcal{O}(n^6)$	[RE99]
Unconstrained	NP-complete	[LP00]

Akutsu/Uemura Algorithm

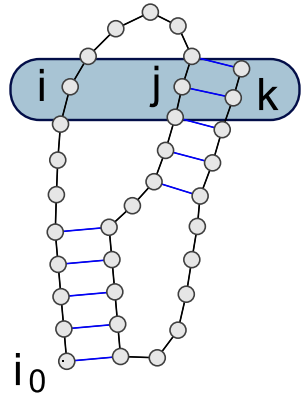
Goal: Capture a category of simple, yet recurrent, pseudoknots.



Idea: When such a PK motif is rotated, one can deduce the MFE of a triplet (i, j, k) from the MFE of triplets directly below it.

Akutsu/Uemura Algorithm

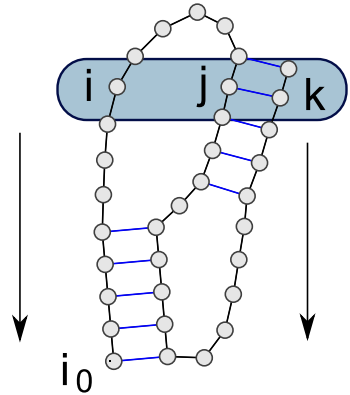
Goal: Capture a category of simple, yet recurrent, pseudoknots.



Idea: When such a PK motif is rotated, one can deduce the MFE of a triplet (i, j, k) from the MFE of triplets directly below it.

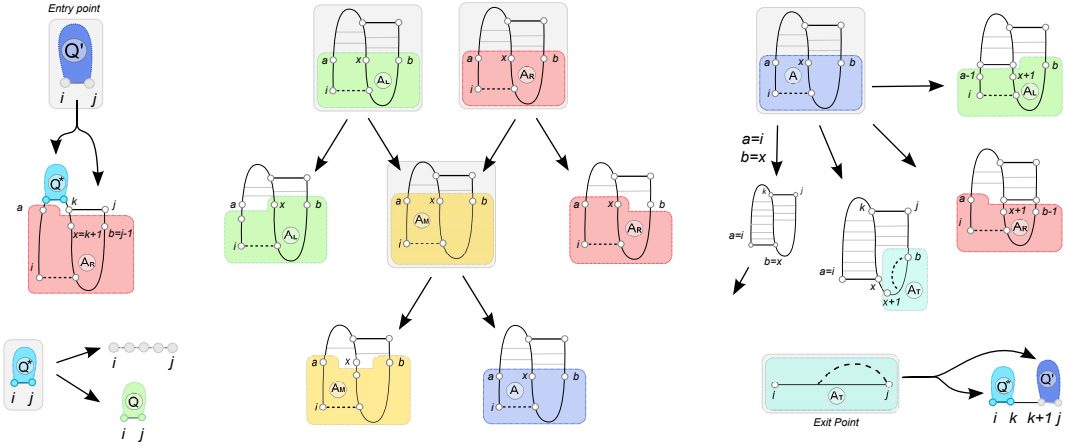
Akutsu/Uemura Algorithm

Goal: Capture a category of simple, yet recurrent, pseudoknots.



Idea: When such a PK motif is rotated, one can deduce the MFE of a triplet (i, j, k) from the MFE of triplets directly below it.

Akutsu/Uemura: Dynamic programming



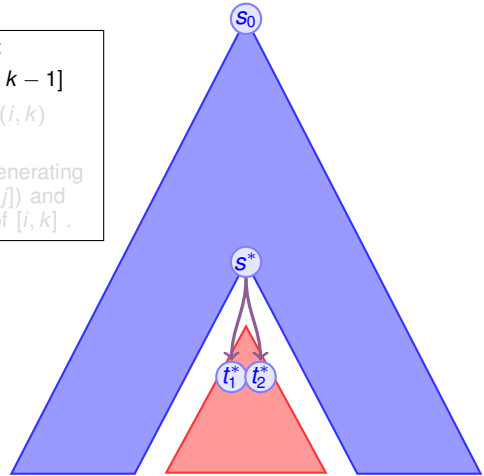
Application/Problem	Weight fun.	Time/Space	Ref.
Energy minimization	π_{bp}	$\mathcal{O}(n^4)/\mathcal{O}(n^4)$	[Aku00]
Partition function	$e^{\frac{-\pi_{bp}}{RT}}$	$\mathcal{O}(n^4)/\mathcal{O}(n^4)$	$\Theta(n^6)$[CC09]
BP probabilities	$e^{\frac{-\pi_{bp}}{RT}}$	$\mathcal{O}(n^4)/\mathcal{O}(n^4)$	–
Sampling (k-struct.)	$e^{\frac{-\pi_{bp}}{RT}}$	$\mathcal{O}(n^4 + kn\log n)/\mathcal{O}(n^4)$	–

Exercice: Write DP equation for MFE computation, counting and partition function.

Inside/outside algorithm

Structure including base pair (i, k):

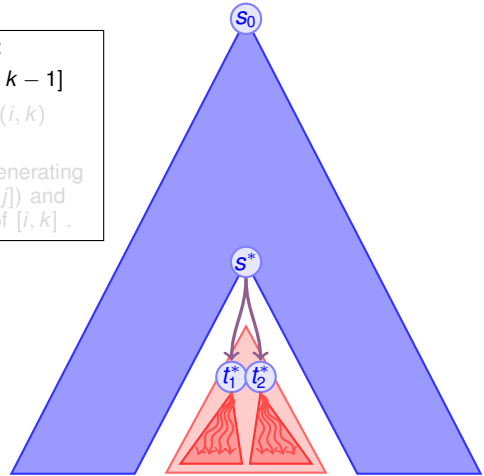
- Inside: Structures over $[i+1, k-1]$
- Outside: Contexts of interval (i, k)
  - $\forall$ interval $[i, j], i < j \leq k$
  - Complete structure by generating brother intervals ($[k+1, j]$) and recurse over the father of $[i, k]$.



Inside/outside algorithm

Structure including base pair (i, k):

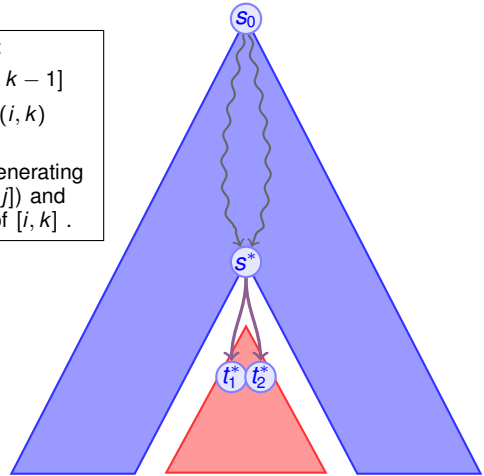
- Inside: Structures over $[i+1, k-1]$
- Outside: Contexts of interval (i, k)
  - $\forall$ interval $[i, j], i < j \leq k$
  - Complete structure by generating brother intervals ($[k+1, j]$) and recurse over the father of $[i, k]$.



Inside/outside algorithm

Structure including base pair (i, k):

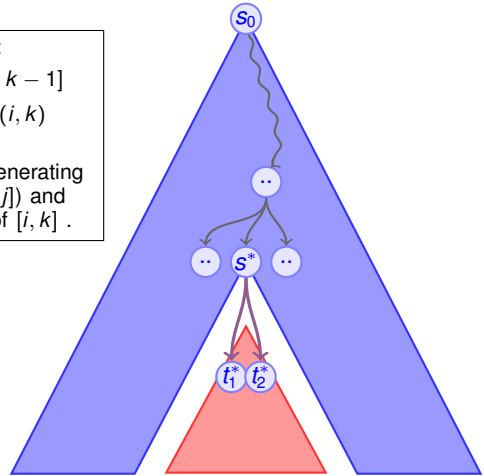
- Inside: Structures over $[i+1, k-1]$
- Outside: Contexts of interval (i, k)
  - $\forall$ interval $[i, j], i < j \leq k$
  - Complete structure by generating brother intervals ($[k+1, j]$) and recurse over the father of $[i, k]$.



Inside/outside algorithm

Structure including base pair (i, k):

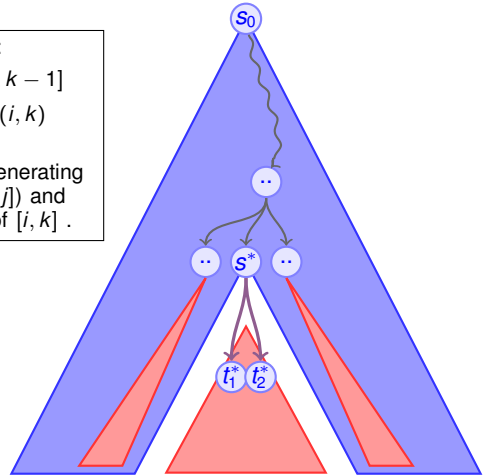
- Inside: Structures over $[i+1, k-1]$
- Outside: Contexts of interval (i, k)
  - $\forall$ interval $[i, j], i < j \leq k$
  - Complete structure by generating brother intervals ($[k+1, j]$) and recurse over the father of $[i, k]$.



Inside/outside algorithm

Structure including base pair (i, k):

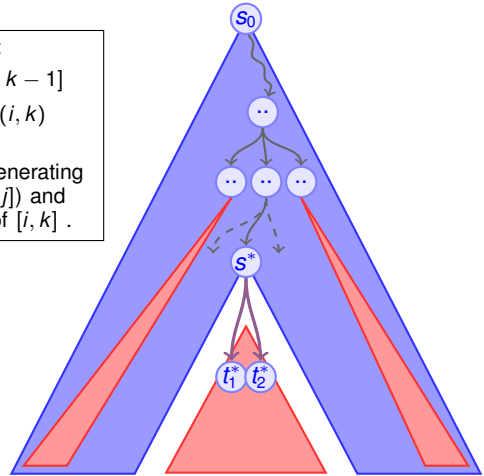
- Inside: Structures over $[i+1, k-1]$
- Outside: Contexts of interval (i, k)
  - $\forall$ interval $[i, j], i < j \leq k$
  - Complete structure by generating brother intervals ($[k+1, j]$) and recurse over the father of $[i, k]$.



Inside/outside algorithm

Structure including base pair (i, k):

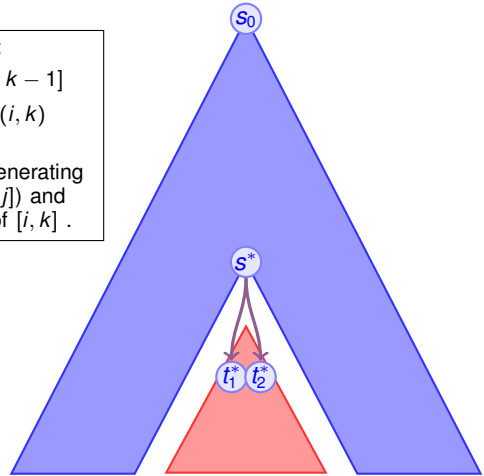
- Inside: Structures over $[i+1, k-1]$
- Outside: Contexts of interval (i, k)
  - $\forall$ interval $[i, j], i < j \leq k$
  - Complete structure by generating brother intervals ($[k+1, j]$) and recurse over the father of $[i, k]$.



Inside/outside algorithm

Structure including base pair (i, k):

- Inside: Structures over $[i+1, k-1]$
- Outside: Contexts of interval (i, k)
  - $\forall$ interval $[i, j], i < j \leq k$
  - Complete structure by generating brother intervals ($[k+1, j]$) and recurse over the father of $[i, k]$.



Whenever some further technical conditions are satisfied, this decomposition is complete and unambiguous, and implies a *simple recurrence* for computing the base pair probability matrix in $\Theta(n^3)$.

Alternatively: Duplicate sequence

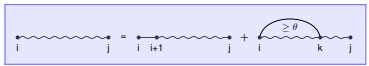
What is a good dynamic programming scheme?

Correction of a (Ensemble) dynamic programming scheme:

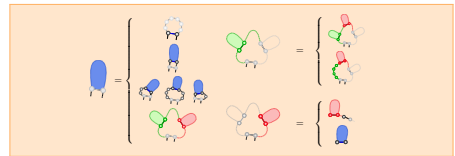
Objective function correctly computed/inherited at local level

\+ All the conformations can be obtained

$\Rightarrow$ Correct algorithm (Induction)



?



Enumerating search space helps but does not constitute a proof.

Need to show equivalence of DP schemes, *e.g.* use one to simulate the other and vice versa.

(Generating functions may help)

What is a good dynamic programming scheme?

Correction of a (Ensemble) dynamic programming scheme:

Objective function correctly computed/inherited at local level

\+ All the conformations can be obtained

$\Rightarrow$ Correct algorithm (Induction)

$$C_{i,t} = 1, \quad \forall t \in [i, i+\theta]$$
$$C_{i,j} = \sum \left\{ \sum_{k=i+\theta+1}^{j} 1 \times C_{i+1,k-1} \times C_{k+1,j} \right.^{C_{i+1,j}}$$

Homopolymer (All pairs allowed) + $\theta = 1$
$\Rightarrow C_{1,n} = \mathbf{1}, 1, 1, 2, 4, 8, 17, 32, 82, 185, 423, \ldots$

$\Leftrightarrow$?

$$C'_{i,j} = \sum \left\{ \begin{array}{l} 1 \\ C'_{i+1,j-1} \\ \sum_{i',j'} C'_{i',j'} \\ \sum_k C_{i+1,k-1} \times C^1{}_{k,j-1} \end{array} \right.$$
$$C_{i,j} = \sum_k \left((C_{i,k-1} + 1) \times C^1{}_{k,j} \right)$$
$$C^1{}_{i,j} = C^1{}_{i,j-1} + C'_{i,j}$$

Homopolymer + $\theta = 1$
$\Rightarrow C'_{1,n} = \mathbf{0}, 1, 1, 2, 4, 8, 17, 32, 82, 185, 423, \ldots$

Enumerating search space helps but does not constitute a proof.

Need to show equivalence of DP schemes, *e.g.* use one to simulate the other and vice versa.

(Generating functions may help)

References I


Tatsuya Akutsu.
Dynamic programming algorithms for rna secondary structure prediction with pseudoknots.
Discrete Appl. Math., 104(1-3):45–62, 2000.


S. Cao and S-J Chen.
Predicting structured and stabilities for h-type pseudoknots with interhelix loop.
RNA, 15:696–706, 2009.


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.

References II


Robert M Dirks and Niles A Pierce.
A partition function algorithm for nucleic acid secondary structure including pseudoknots.
J Comput Chem, 24(13):1664–1677, Oct 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.


D. H. Mathews, J. Sabina, M. Zuker, and D. H. Turner.
Expanded sequence dependence of thermodynamic parameters improves prediction of rna secondary structure.
Journal of Molecular Biology, 288(5):911–940, May 1999.

References III


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.


E. Rivas and S.R. Eddy.
A dynamic programming algorithm for RNA structure prediction including pseudoknots.
J Mol Biol, 285:2053–2068, 1999.


S. Wuchty, W. Fontana, I.L. Hofacker, and P. Schuster.
Complete suboptimal folding of RNA and the stability of secondary structures.
Biopolymers, 49:145–164, 1999.