The Ensemble of RNA Structures

Example: best structures of the RNA sequence

```
GGGGGUAUAGCUCAGGGGUAGAGCAUUUGACUGCAGAUCAAGAGGUCCCUGGUUCAAAUCCAGGUGCCCCCU
                                                                 free energy in kcal/mol
((((((..((((.......))))..........((((....))))((((.......))))))))))). -28.10
((((((..((((.......))))....((((.(.......).))))((((.......))))))))))). -27.90
(((((((.((((.......))))(((((((..(((....))))..))))..)))))....))))))). -27.80
(((((((.((((.......))))(((((((..(((....))))..)))..)))))....))))))). -27.80
((((((..((((.......))))....(((..........))))((((.......))))))))))). -27.60
((((((..((((.......))))....((..(.......)..))((((.......))))))))))). -27.50
(((((((.((((.......)))).(((((((..(((....))))..))))..))))....))))))). -27.20
(((((((.((((.......)))).(((((((..(((....))))..)))..)))))....))))))). -27.20
(((((((.((((.......))))..........(((....)))).((((.......))))))))))). -27.20
((((((...((((.......))))..........(((....))))((((.......)))))..)))))). -27.20
((((((...(((...(((...(((......)))..)))..)))...((((.......))))))))))). -27.10
(((((((.((((.......))))(((((((...(((....))))...)))..)))))....))))))). -27.00
(((((((.((((.......))))(((((((...(((....))))...))..)))))....))))))). -27.00
((((((..((((.......))))....((((.(.......).))))..((((.......))))))))))). -27.00
((((((..((((.......)))).(((((....).)))))..((((.......))))))))))). -27.00
((((((..((((.......))))..........((......))((((.......))))))))))). -27.00
((((((...((((.......))))....((((.(.......).))))((((.......)))))..)))))). -27.00
(((((((.((((.......))))(((((((..(((......)))..))))..)))))....))))))). -26.70
(((((((.((((.......))))(((((((..(((......)))..)))..)))))....))))))). -26.70
(((((((.((((.......))))....(((..........))))..((((.......))))))))))). -26.70
((((((..((((.......)))).((((.......))))).....((((.......))))))))))). -26.70
((((((...((((.......))))....(((..........))))((((.......)))))..)))))). -26.70
```

The set of all non-crossing RNA structures of an RNA sequence S is called *(structure) ensemble $\mathcal{P}$ of S*.

Is Minimal Free Energy Structure Prediction Useful?

- BIG PLUS: loop-based energy model quite realistic
- Still mfe structure may be "wrong": Why?
- Lesson: be careful, be sceptical!
  (as always, but in particular when biology is involved)
- What would you improve?

Probability of a Structure

How probable is an RNA structure P for a RNA sequence S?
GOAL: define probability $\Pr[P|S]$.
IDEA: Think of RNA folding as a *dynamic system* of structures (=states of the system). Given much time, a sequence S will form every possible structure P. For each structure there is a probability for observing it at a given time.

This means: we look for a probability distribution!
Requirements: probability depends on energy — the lower the more probable. No additional assumptions!

Distribution of States in a System

Definition (Boltzmann distribution)

Let $\mathcal{X} = \{X_1, \ldots, X_N\}$ denote a system of states, where state X_i has energy E_i. The system is *Boltzmann distributed with temperature T* iff $\Pr[X_i] = exp(-\beta E_i)/Z$ for $Z := \sum_i exp(-\beta E_i)$, where $\beta = (k_B T)^{-1}$.

Remarks

- broadly used in physics to describe systems of whatever
- Boltzmann distribution is usually assumed for the *thermodynamic equilibrium* (i.e. after sufficiently much time)
- transfer to RNA easy to see: structures=states, energies
- why temperature?
  - very high temperature: all states equally probable
  - very low temperature: only best states occur
- $k_B \approx 1.38 \times 10^{-23} J/K$ is known as *Boltzmann constant*; β is called *inverse temperature*.
- call $exp(-\beta E_i)$ *Boltzmann weight of X_i*.

What next?

We assume that the structure ensemble of an RNA sequence is Boltzmann distributed.

- What are the benefits?
  (More than just probabilities of structures . . .)
- Why is it reasonable to assume Boltzmann distribution?
  (Well, a physicist told me . . .)
- How to calculate probabilities efficiently?
  (McCaskill's algorithm)

Benefits of Assuming Boltzmann

Definition

Probability of a structure P for S: $\Pr[P|S] := exp(-\beta E(P))/Z$.

Allows more profound weighting of structures in the ensemble. We need efficient computation of partition function Z!

Even more interesting: probability of structural elements

Definition

Probability of a base pair (i,j) for S:

$$\Pr[(i,j)|S] := \sum_{P \ni (i,j)} \Pr[P|S]$$

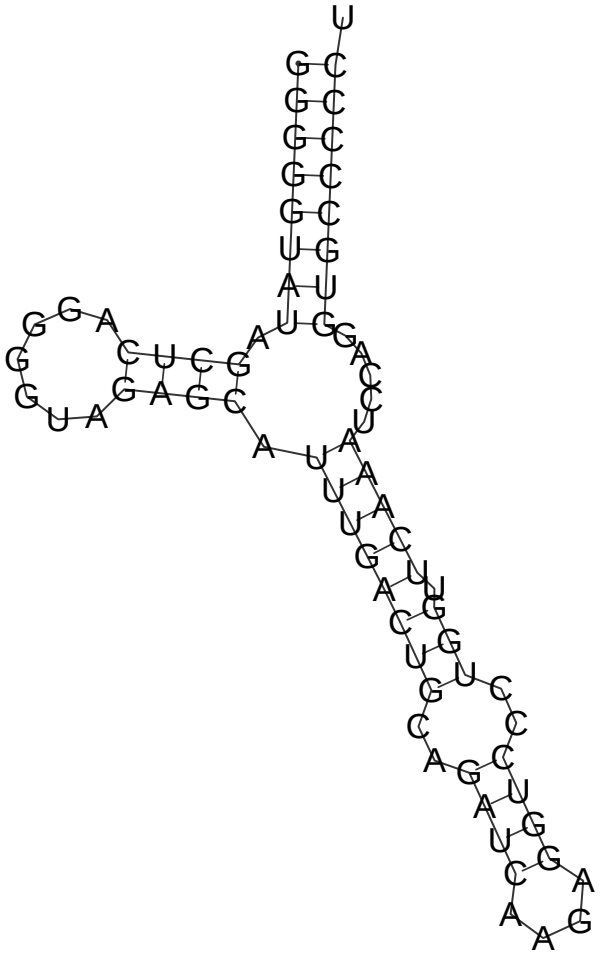
Again, we need Z (and some more). Base pair probabilities enable a new view at the structure ensemble (visually but also algorithmically!).

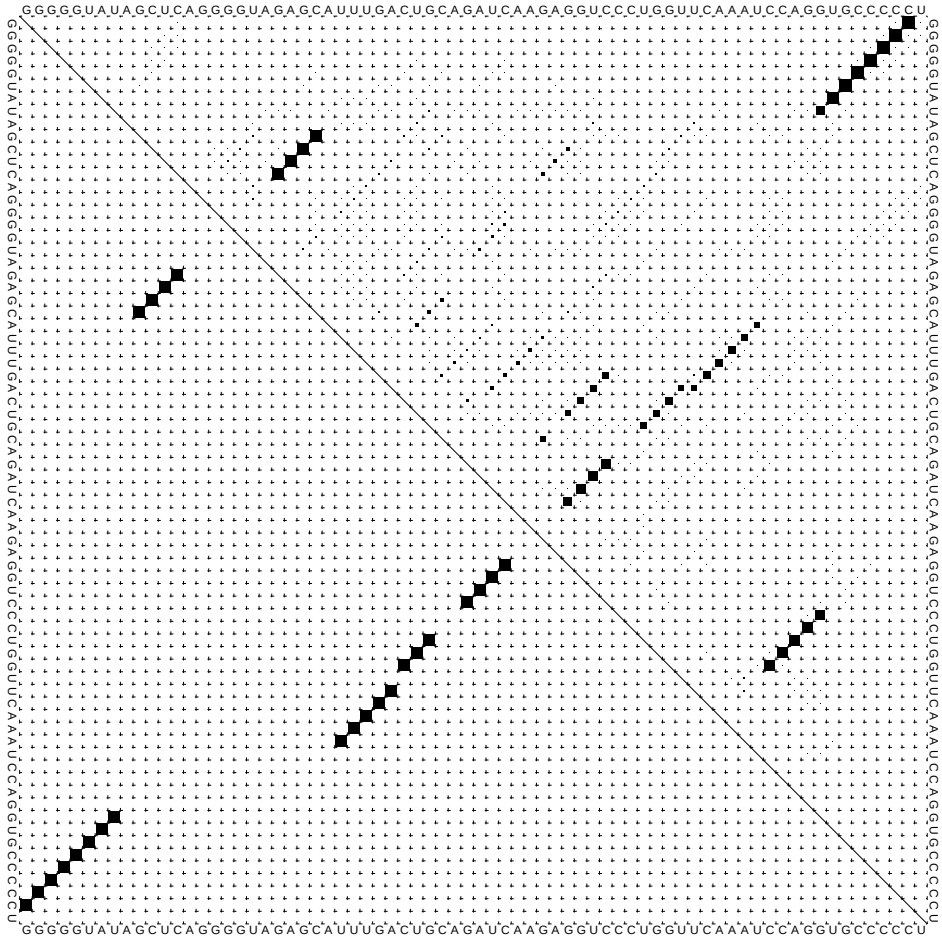
Remark: For RNA, we have "real" temperature, e.g. $T = 37^\circ C$, which determines $\beta = (k_B T)^{-1}$. For calculations pay attention to physical units!

An Immediate Use of Base Pair Probabilities

MFE structure and base pair probability dot plot[1] of a tRNA

GGGGGUAUAGCUCAGGGGUAGAGCAUUUGACUGCAGAUCAAGAGGUCCCUGGUUCAAAUCCAGGUGCCCCCU





[1] computed by "RNAfold -p"

Why Do We Assume Boltzmann

We will give an argument from information theory. We will show: **The Boltzmann distribution makes the least number of assumptions. Formally, the B.d. is the distribution with the lowest information content/maximal (Shannon) entropy.**

As a consequence: without further information about our system, Boltzmann is our best choice.

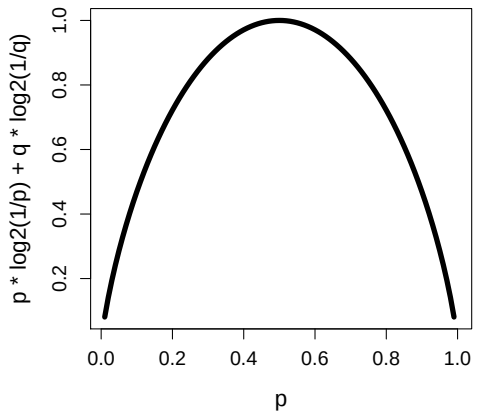
[What could "further information" mean in a biological context?]

Shannon Entropy (by Example)

We toss a coin. For our coin, heads and tails show up with respective probabilities p and q (not necessarily fair).
How uncertain are we about the result?

Answer: expected information

$$H = p \log_b \frac{1}{p} + q \log_b \frac{1}{q}.$$



$p = 0.5, q = 0.5 \Rightarrow$ $H = 1$ — maximal uncertainty

$p = 1, q = 0 \Rightarrow$ $H = 0$ — no uncertainty

This is *Shannon entropy* — a measure of uncertainty.
In general, define the *Shannon entropy*[2] as

$$H(\vec{p}) := -\sum_{i=1}^{N} p_i \log_b p_i.$$

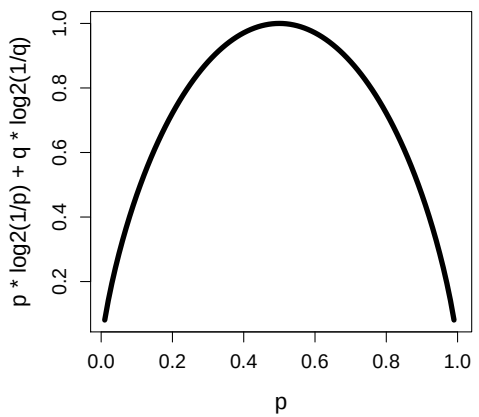
[2] of a probability distribution $\vec{p}$ over N states $X_1 \ldots X_N$

Shannon Entropy (by Example)

We toss a coin. For our coin, heads and tails show up with respective probabilities p and q (not necessarily fair).
How uncertain are we about the result?

Answer: expected information

$$H = p \log_b \frac{1}{p} + q \log_b \frac{1}{q}.$$



$p = 0.5, q = 0.5 \Rightarrow$ $H = 1$ — maximal uncertainty

$p = 1, q = 0 \Rightarrow$ $H = 0$ — no uncertainty

This is *Shannon entropy* — a measure of uncertainty.
In general, define the *Shannon entropy*[2] as

$$H(\vec{p}) := -\sum_{i=1}^{N} p_i \log_b p_i.$$

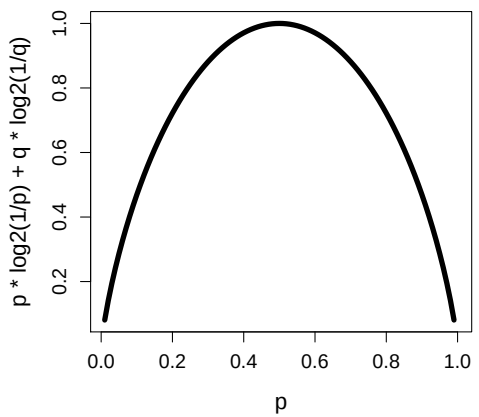
[2] of a probability distribution $\vec{p}$ over N states $X_1 \ldots X_N$

Shannon Entropy (by Example)

We toss a coin. For our coin, heads and tails show up with respective probabilities p and q (not necessarily fair).
How uncertain are we about the result?

Answer: expected information

$$H = p \log_b \frac{1}{p} + q \log_b \frac{1}{q}.$$



$p = 0.5, q = 0.5 \Rightarrow$ $H = 1$ — maximal uncertainty

$p = 1, q = 0 \Rightarrow$ $H = 0$ — no uncertainty

This is *Shannon entropy* — a measure of uncertainty.
In general, define the *Shannon entropy*[2] as

$$H(\vec{p}) := -\sum_{i=1}^{N} p_i \log_b p_i.$$

[2] of a probability distribution $\vec{p}$ over N states $X_1 \ldots X_N$

Formalizing "Least number of assumptions"

Example:
Assume: we have N events. Without further assumptions, we will naturally assume the uniform distribution

$$p_i = \frac{1}{N}.$$

This is the uniquely defined distribution maximizing the entropy $H(\vec{p}) = -\sum_i p_i \log_b p_i$.
It is found by solving the following optimization problem:

maximize the function

$$H(\vec{p}) = -\sum_i p_i \log_b p_i$$

under the side condition $\sum_i p_i = 1$.

Formalizing "Least number of assumptions"

Theorem: Given a system of states $X_1 \ldots X_N$ and energies E_i for X_i. The Boltzmann distribution is the probability distribution $\vec{p}$ that maximizes Shannon entropy

$$H(\vec{p}) = -\sum_{i=1}^{N} p_i \log_b p_i$$

under the assumption of known average energy of the system

$$< E >= \sum_{i=1}^{N} p_i E_i.$$

Proof

We show that the Boltzmann distribution is uniquely obtained by solving

$$\textit{maximize function } H(\vec{p}) = -\sum_{i=1}^{N} p_i \ln p_i \qquad [3]$$

under the side conditions

- $C_1(\vec{p}) = \sum_i p_i - 1 \quad = 0$ *and*
- $C_2(\vec{p}) = \sum_i p_i E_i - < E > \quad = 0$

by using the method of Lagrange multipliers.

[3] whether using ln or $\log_b$ is equivalent for maximization

Proof Using Lagrange Multipliers

Following the trick of Lagrange, find the extreme value of

$$L(\vec{p}, \alpha, \beta) = H(\vec{p}) - \alpha C_1(\vec{p}) - \beta C_2(\vec{p}).$$

By construction, $C_1(\vec{p})$ and $C_2(\vec{p})$ are partial derivatives:

$$\frac{\partial L(\vec{p}, \alpha, \beta)}{\partial \alpha} = C_1(\vec{p})$$
$$\frac{\partial L(\vec{p}, \alpha, \beta)}{\partial \beta} = C_2(\vec{p})$$

Thus the side conditions hold at the optimum, since there all partial derivatives are 0.

Proof (Ctd.) — Partial Derivatives w.r.t p_j

Futhermore, we need the partial derivatives with respect to p_j

$$\begin{aligned}\frac{\partial L(\vec{p},\alpha,\beta)}{\partial p_j} &= \frac{\partial H(\vec{p})}{\partial p_j} - \alpha\frac{\partial C_1(\vec{p})}{\partial p_j} - \beta\frac{\partial C_2(\vec{p})}{\partial p_j} \\ &= -\frac{\partial \sum_{i=1}^{N} p_i \ln p_i}{\partial p_j} - \alpha\frac{\partial \sum_i p_i - 1}{\partial p_j} - \beta\frac{\partial \sum_i p_i E_i - < E >}{\partial p_j} \\ &= -(\ln p_j + 1) - \alpha - \beta E_j\end{aligned}$$

Proof (Ctd.) — Solve Equations

Finally, we need to solve the system

$$\sum_i p_i E_i - < E > = 0 \tag{1}$$

$$\sum_i p_i - 1 = 0 \tag{2}$$

$$-(\ln p_j + 1) - \alpha - \beta E_j = 0 \tag{3}$$

Remarks

- Resolving (3) to p_j and putting into (2) yields a distribution of the same form as the Boltzmann distribution.
- We won't show the dependency of $\beta = k_B T^{-1}$ and $< E >$.

Proof (Ctd)

Equation (3) can be rewritten to:

$$\ln p_j = -\beta E_j - (\alpha + 1).$$

Thus by exponentiation on both sides

$$p_j = \exp(-\beta E_j - \gamma) = \frac{\exp(-\beta E_j)}{\exp(\gamma)}, \tag{4}$$

where $\gamma = (\alpha + 1)$.
By substituting (4) in (2) $\sum_i p_i - 1 = 0$ we get

$$1 = \sum_i \exp(-\beta E_j)/\exp(\gamma) \quad \text{and thus} \quad \exp(\gamma) = \sum_i \exp(-\beta E_i)$$

□

Partition Function

Recall: For probabilities, $\Pr[P|S] = \exp(-\beta E(P))/Z$, we need Z.

Definition

For an RNA sequence S, we call

$$Z := \sum_{P \text{ non-crossing RNA structure for } S} \exp(-\beta E(P))$$

the *partition function (of the RNA ensemble $\mathcal{P}$) of S*.

Remark

Naive computation of Z: exponential, since ensemble size is exponential in $|S|$.

Excursion: Counting of Structures

Problem of computing the partition function is similar to counting the structures in the ensemble $\mathcal{P}$. Partition function is a weighted sum, in counting we "weight" structures by 1.

How to count non-crossing RNA structures for S?

Example: S=CGAGC (minimal loop length m=0).

- naïve: enumerate $\Rightarrow$ exponential
- efficient: DP with decomposition a la Nussinov

Excursion: Counting of Structures

Problem of computing the partition function is similar to counting the structures in the ensemble $\mathcal{P}$. Partition function is a weighted sum, in counting we "weight" structures by 1.

How to count non-crossing RNA structures for S?

Example: S=CGAGC (minimal loop length m=0).

- naïve: enumerate $\Rightarrow$ exponential
- efficient: DP with decomposition a la Nussinov

Enumerating Structures: S=CGAGC

C_1	G_2	A_3	G_4	C_5	
					C_1
					G_2
					A_3
					G_4
					C_5

Enumerating Structures: S=CGAGC

C_1	G_2	A_3	G_4	C_5	
{.}	{..,()}	{...,().}	{....,()..,(..)}	{.....,()...,(..)., .(..),....(),().()}	C_1
	{.}	{..}	{...}	{....,(..)}	G_2
		{.}	{..}	{...,.()}	A_3
			{.}	{..,()}	G_4
				{.}	C_5

Subensembles

Definition (Subensemble)

Define the *ij-subensemble $\mathcal{P}_{ij}$ of S* (for $1 \leq i \leq j \leq n$) as

$$\mathcal{P}_{ij} := \text{set of all non-crossing RNA } ij\text{-substructures } P \text{ of } S.$$

where:

Definition (RNA Substructure)

An RNA structure P of S is called *ij-substructure of S* iff $P \subseteq \{i, \ldots, j\}^2$.

Remarks

- Example: see last slide, $\mathcal{P}_{14} = \{\{\}, \{(1,2)\}, \{(1,4)\}\}$,
  $\mathcal{P}_{15} = \{\{\}, \{(1,2)\}, \{(1,4)\}, \{(2,5)\}, \{(4,5)\}, \{(1,2),(4,5)\}\}$
- ensemble $\mathcal{P}$ of S: $\mathcal{P} = \mathcal{P}_{1n}$
- $\mathcal{P}_{ij} = \{\{\}\}$ for $j < i + m$ (min. loop size m)

Efficient Counting of Structures

Define: $C_{ij} := |\mathcal{P}_{ij}|$. ($\Rightarrow$ DP-matrix C)

Computation of C_{ij}

for $j - i \leq m$: $C_{ij} = 1$, since $\mathcal{P}_{ij} = \{\{\}\}$

for $j - i > m$: recurse!

$\mathcal{P}_{ij}$ consists of structures

$$\mathcal{P}_{ij-1} \qquad (\ j \text{ unpaired})$$

and structures

$$\mathcal{P}_{ik-1} \otimes \mathcal{P}_{k+1j-1} \otimes \{\{(k,j)\}\} \qquad (\ k,j \text{ paired }),$$

where:

"$\otimes$" combines all structures in one set with all structures in a second set.

Define: $\mathcal{P} \otimes \mathcal{Q} := \{P \cup Q | P \in \mathcal{P}, Q \in \mathcal{Q}\}$.

Efficient Counting of Structures

Define: $C_{ij} := |\mathcal{P}_{ij}|$. ($\Rightarrow$ DP-matrix C)

Computation of C_{ij}

for $j - i \leq m$: $C_{ij} = 1$, since $\mathcal{P}_{ij} = \{\{\}\}$

for $j - i > m$: recurse!

$\mathcal{P}_{ij}$ consists of structures

$$\mathcal{P}_{ij-1} \qquad (\ j \text{ unpaired})$$

and structures

$$\mathcal{P}_{ik-1} \otimes \mathcal{P}_{k+1j-1} \otimes \{\{(k,j)\}\} \qquad (\ k,j \text{ paired }),$$

where:

"$\otimes$" combines all structures in one set with all structures in a second set.

Define: $\mathcal{P} \otimes \mathcal{Q} := \{P \cup Q | P \in \mathcal{P}, Q \in \mathcal{Q}\}$.

Computation of C_{ij}

for $j - i > m$:

$$\mathcal{P}_{ij} = \mathcal{P}_{ij-1} \cup \bigcup_{\substack{i \le k < j-m \\ S_k, S_j \text{ compl.}}} \mathcal{P}_{ik-1} \otimes \mathcal{P}_{k+1j-1} \otimes \{\{(k,j)\}\}$$

this means for C_{ij}: recall $C_{ij} = |\mathcal{P}_{ij}|$

$$C_{ij} = C_{ij-1} + \sum_{\substack{i \le k < j-m \\ S_k, S_j \text{ compl.}}} C_{ik-1} \cdot C_{k+1j-1} \cdot 1$$

Remarks

- by DP: compute ensemble size C_{1n} in $O(n^3)$ time and $O(n^2)$ space.
- why "translates" $\cup$ to $+$ and $\otimes$ to $\cdot$? $\Leftarrow$ all unions were disjoint!
  i.e.: 1.) cases in "$\mathcal{P}_{ij}$ consists of ..." are disjoint
  2.) structures combined by $\otimes$ are disjoint

Example

decompose sequence $S_{15} = C_1 G_2 A_3 G_4 C_5$

1. subsequence $C_1 G_2 A_3 G_4$ and C_5 unpaired
   $C_{15} \leftarrow C_{14}$
2. a.) k=2. C_1, $A_3 G_4$, base pair $(2, 5)$
   $\mathcal{P}_{15} \leftarrow \mathcal{P}_{11} \otimes \mathcal{P}_{34} \otimes \{\{(2, 5)\}\}$
   $C_{15} \leftarrow C_{11} \cdot C_{34} \cdot 1$
   b.) k=4. $C_1 G_2 A_3$, base pair $(4, 5)$
   $\mathcal{P}_{15} \leftarrow \mathcal{P}_{13} \otimes \mathcal{P}_{54} \otimes \{\{(4, 5)\}\}$
   $C_{15} \leftarrow C_{13} \cdot C_{54} \cdot 1$

ad 2b.)

$\mathcal{P}_{13} \otimes \mathcal{P}_{54} \otimes \{\{(4, 5)\}\} = \{\{\}, \{(1, 2)\}\} \otimes \{\{\}\} \otimes \{\{(4, 5)\}\}$
$= \{\{(4, 5)\}, \{(1, 2), (4, 5)\}\}$

Example

decompose sequence S_{15} =$C_1G_2A_3G_4C_5$

1. subsequence $C_1G_2A_3G_4$ and C_5 unpaired
   $C_{15} \leftarrow C_{14}$
2. a.) k=2. C_1, A_3G_4, base pair $(2,5)$
   $\mathcal{P}_{15} \leftarrow \mathcal{P}_{11} \otimes \mathcal{P}_{34} \otimes \{\{(2,5)\}\}$
   $C_{15} \leftarrow C_{11} \cdot C_{34} \cdot 1$
   b.) k=4. $C_1G_2A_3$, base pair $(4,5)$
   $\mathcal{P}_{15} \leftarrow \mathcal{P}_{13} \otimes \mathcal{P}_{54} \otimes \{\{(4,5)\}\}$
   $C_{15} \leftarrow C_{13} \cdot C_{54} \cdot 1$

ad 2b.)

$\mathcal{P}_{13} \otimes \mathcal{P}_{54} \otimes \{\{(4,5)\}\} = \{\{\}, \{(1,2)\}\} \otimes \{\{\}\} \otimes \{\{(4,5)\}\}$
$= \{\{(4,5)\}, \{(1,2),(4,5)\}\}$

Counting vs. Structure Prediction

Counting

init $C_{ij} = 1 \qquad (j - i \leq m)$

recurse $C_{ij} = C_{ij-1} + \sum_{\substack{i \leq k < j-m \\ S_k, S_j \text{ compl.}}} C_{ik-1} \cdot C_{k+1j-1} \cdot 1$

Prediction

init $N_{ij} = 0 \qquad (j - i \leq m)$

recurse $N_{ij} = \max\{N_{ij-1}, \max_{\substack{i \leq k < j-m \\ S_k, S_j \text{ compl.}}} N_{ik-1} + N_{k+1j-1} + 1\}$

Remarks

- "translation" Prediction $\rightarrow$ Counting : $\max \rightarrow +$, $+ \rightarrow \cdot$
- only possible since sets disjoint, i.e.
  - disjoint cases (no "ambiguity")
  - non-overlapping decomposition in each single case

Back to Computing the Partition Function

Recall: For probabilities, $\Pr[P|S] = \exp(-\beta E(P))/Z$, we need Z.
We defined: $Z := \sum_{P \in \mathcal{P}} \exp(-\beta E(P))$
We claimed: Problem of computing the partition function is similar to counting the structures in the ensemble $\mathcal{P}$. Partition function is a weighted sum, in counting we "weight" structures by 1.

Definition (Partition Function of a Set of Structures)

In analogy to $C_{ij} = |\mathcal{P}_{ij}| = \sum_{P \in \mathcal{P}_{ij}} 1$, define the *partition function $Z_{\mathcal{P}}$ for the set of RNA structures $\mathcal{P}$ of S* by

$$Z_{\mathcal{P}} := \sum_{P \in \mathcal{P}} exp(-\beta E(P)).$$

Idea: compute the $Z_{\mathcal{P}_{ij}}$ recursively $\Rightarrow$ efficient by DP.

Disjoint Decomposition — when to add?

Definition (Disjoint Sets)

Two sets of RNA structures $\mathcal{P}_1$ and $\mathcal{P}_2$ are *(structurally) disjoint* iff $\mathcal{P}_1 \cap \mathcal{P}_2 = \{\}$.

Proposition (Disjoint Decomposition)

Let $\mathcal{P}$, $\mathcal{P}_1$, and $\mathcal{P}_2$ be sets of structures of an RNA sequence S. If $\mathcal{P}_1$ and $\mathcal{P}_2$ are structurally disjoint and $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$, then

$$Z_{\mathcal{P}} = Z_{\mathcal{P}_1} + Z_{\mathcal{P}_2}.$$

Proof

Proof.

$$
\begin{aligned}
Z_{\mathcal{P}} &= \sum_{P \in \mathcal{P}} \exp(-\beta E(P)) \\
&=_{\text{disjoint}} \sum_{P \in \mathcal{P}_1 \uplus \mathcal{P}_2} \exp(-\beta E(P)) \\
&= \sum_{P \in \mathcal{P}_1} \exp(-\beta E(P)) + \sum_{P \in \mathcal{P}_2} \exp(-\beta E(P)) \\
&= Z_{\mathcal{P}_1} + Z_{\mathcal{P}_2}
\end{aligned}
$$

□

Independent Decomposition — when to multiply?

Definition (Independent Sets)

Let S be an RNA sequence. Two sets of non-crossing RNA structures $\mathcal{P}_1$ and $\mathcal{P}_2$ for S are *structurally independent* iff for all $P_1 \in \mathcal{P}_1$ and $P_2 \in \mathcal{P}_2$

1. $P_1 \cap P_2 = \{\}$.
2. each loop/secondary structure element of the RNA structure $P = P_1 \cup P_2$ is either a loop of P_1 or one of P_2.

Proposition (Independent Decomposition)

Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be structurally independent sets of non-crossing RNA structures for RNA sequence S and $\mathcal{P} = \mathcal{P}_1 \otimes \mathcal{P}_2$. Then:

$$Z_{\mathcal{P}} = Z_{\mathcal{P}_1} \cdot Z_{\mathcal{P}_2}$$

Remark: Condition (1) suffices for energy functions based on scoring base pairs (like in Nussinov). For loop-based energy models, we need (2), which implies $E(P_1 \cup P_2) = E(P_1) + E(P_2)$.

Proof

Proof.

$$
\begin{aligned}
Z_{\mathcal{P}} &= \sum_{P \in \mathcal{P}} \exp(-\beta E(P)) \\
&=_{\textit{indep.}(1)} \sum_{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2} \exp(-\beta E(P_1 \cup P_2)) \\
&=_{\textit{indep.}(2)} \sum_{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2} \exp(-\beta (E(P_1) + E(P_2))) \\
&= \sum_{P_1 \in \mathcal{P}_1} \sum_{P_2 \in \mathcal{P}_2} \exp(-\beta E(P_1)) \exp(-\beta E(P_2)) \\
&= \sum_{P_1 \in \mathcal{P}_1} \exp(-\beta E(P_1)) \left(\sum_{P_2 \in \mathcal{P}_2} \exp(-\beta E(P_2)) \right) \\
&= \sum_{P_1 \in \mathcal{P}_1} \exp(-\beta E(P_1)) Z_{\mathcal{P}_2} \\
&= Z_{\mathcal{P}_1} \cdot Z_{\mathcal{P}_2}
\end{aligned}
$$

Adding and Multiplying of Partition Functions

in the same way as for counts!

Counting

init $C_{ij} = 1 \qquad (j - i \leq m)$

recurse $C_{ij} = C_{ij-1} + \sum_{\substack{i \leq k < j-m \\ S_k, S_j \text{ compl.}}} C_{ik-1} \cdot C_{k+1j-1} \cdot 1$

Partition Function

init $Z_{\mathcal{P}_{ij}} = 1 \qquad (j - i \leq m)$

recurse

$Z_{\mathcal{P}_{ij}} = Z_{\mathcal{P}_{ij-1}} + \sum_{\substack{i \leq k < j-m \\ S_k, S_j \text{ compl.}}} Z_{\mathcal{P}_{ik-1}} \cdot Z_{\mathcal{P}_{k+1j-1}} \cdot \exp(-\beta \text{“}E(basepair)\text{”})$

Remarks

- “E(basepair)”: e.g. -1 or depending on S_i and S_j for base pair (i, j)
- This partitition function variant of the Nussinov algorithm can **not** compute the partition function for the loop-based energy model(!)

Adding and Multiplying of Partition Functions

in the same way as for counts!

Counting

init $C_{ij} = 1 \qquad (j - i \leq m)$

recurse $C_{ij} = C_{ij-1} + \sum_{\substack{i \leq k < j-m \\ S_k, S_j \text{ compl.}}} C_{ik-1} \cdot C_{k+1j-1} \cdot 1$

Partition Function

init $Z_{\mathcal{P}_{ij}} = 1 \qquad (j - i \leq m)$

recurse

$Z_{\mathcal{P}_{ij}} = Z_{\mathcal{P}_{ij-1}} + \sum_{\substack{i \leq k < j-m \\ S_k, S_j \text{ compl.}}} Z_{\mathcal{P}_{ik-1}} \cdot Z_{\mathcal{P}_{k+1j-1}} \cdot \exp(-\beta \text{“}E(\text{basepair})\text{”})$

Remarks

- “E(basepair)”: e.g. -1 or depending on S_i and S_j for base pair (i, j)
- This partitition function variant of the Nussinov algorithm can **not** compute the partition function for the loop-based energy model(!)

Adding and Multiplying of Partition Functions

in the same way as for counts!

Counting

init $C_{ij} = 1 \qquad (j - i \leq m)$

recurse $C_{ij} = C_{ij-1} + \sum_{\substack{i \leq k < j-m \\ S_k, S_j \text{ compl.}}} C_{ik-1} \cdot C_{k+1j-1} \cdot 1$

Partition Function

init $Z^N_{\mathcal{P}_{ij}} = 1 \qquad (j - i \leq m)$

recurse

$$Z^N_{\mathcal{P}_{ij}} = Z^N_{\mathcal{P}_{ij-1}} + \sum_{\substack{i \leq k < j-m \\ S_k, S_j \text{ compl.}}} Z^N_{\mathcal{P}_{ik-1}} \cdot Z^N_{\mathcal{P}_{k+1j-1}} \cdot \exp(-\beta \text{“}E(basepair)\text{”})$$

Remarks

- “E(basepair)”: e.g. -1 or depending on S_i and S_j for base pair (i, j)
- This partitition function variant of the Nussinov algorithm can **not** compute the partition function for the loop-based energy model(!)

Way to RNA Partition Function

- Partition function adding/multiplying like in counting
  Attention: only for disjoint/independent sets
- Loop energy model
  Zuker: how to decompose structure space
  how to compute the energies (as sum of loop energies)

What next?
What is missing?

Way to RNA Partition Function

- Partition function adding/multiplying like in counting
  Attention: only for disjoint/independent sets
- Loop energy model
  Zuker: how to decompose structure space
  how to compute the energies (as sum of loop energies)

What next?

Develop recursions for partition function using "real" RNA energies

Plan: rewrite Zuker-algo into its partition function variant

What is missing?

Way to RNA Partition Function

- Partition function adding/multiplying like in counting
  Attention: only for disjoint/independent sets
- Loop energy model
  Zuker: how to decompose structure space
  how to compute the energies (as sum of loop energies)

What next?

Develop recursions for partition function using "real" RNA energies

Plan: rewrite Zuker-algo into its partition function variant

What is missing?

Is Zuker's decomposition of structure space

- disjoint?
- independent?