ALGORITHMS OF BIOINFORMATICS

8 RNA Structure Prediction

29 January 2026

Prof. Dr. Sebastian Wild

CS594 (Winter 2025-26)
Philipps-Universität Marburg
version 2026-01-29 13:56

Outline

8 RNA Structure Prediction

8.1 Noncoding RNA

RNA

RNA (Ribonucleic acid)

- similar to DNA: polymer of *nucleotides*

⇝ sequence of *nitrogenous bases*
Adenine, and *Cytosine*, *Guanine*, *Uracil*

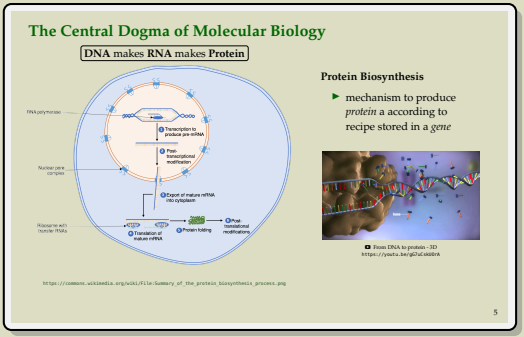
- unlike DNA, typically *single-stranded*
- more “sticky” backbone

RNA

RNA (Ribonucleic acid)

- similar to DNA: polymer of *nucleotides*
- ⇝ sequence of *nitrogenous bases*
  Adenine, and *Cytosine*, *Guanine*, *Uracil*
- unlike DNA, typically *single-stranded*
- more “sticky” backbone

- mostly known as *messenger RNA (mRNA)*
  - including *mRNA vaccines*!
  - mRNA is a coding RNA
    since they encode a protein



Noncoding RNA

But RNA serves many other roles!

Introduction to Non-Coding RNA
https://youtu.be/KIohfQsRRdQ

Noncoding RNA

But RNA serves many other roles!

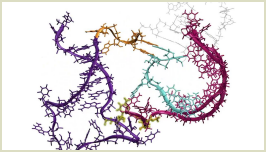
Introduction to Non-Coding RNA
https://youtu.be/KIohfQsRRdQ

- ironically, *ribosomes* (protein factories) themselves are mostly made of RNA
- for noncoding RNA, structure (3D folding form) crucial for function
- indeed, sequence often highly variable between species, but structure is similar!

RNA Secondary Structure Prediction

- Unfortunately, 3D shape hard and expensive to determine experimentally (*X-ray crystallography*)
- Available (diverse) data much smaller than for proteins
  - ⇝ May **not** soon see successful machine-learning solutions similar to AlphaFold

    Rhiju Das, `https://youtu.be/XqFq_zYx7Vo`
- To make matters worse, often not a single static structure

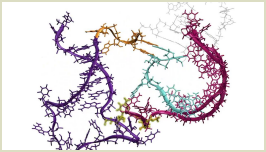


RNA folding in action
`https://youtu.be/2XTi9LG9NnU`

RNA Secondary Structure Prediction

- Unfortunately, 3D shape hard and expensive to determine experimentally (*X-ray crystallography*)
- Available (diverse) data much smaller than for proteins
  - ⇝ May **not** soon see successful machine-learning solutions similar to AlphaFold

    Rhiju Das, `https://youtu.be/XqFq_zYx7Vo`
- To make matters worse, often not a single static structure



RNA folding in action
`https://youtu.be/2XTi9LG9NnU`

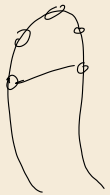
- ⇝ study *de-novo* approaches
- ⇝ and use simplified models of chemistry and shape to make progress

8.2 RNA Secondary Structure

Model of RNA Structure

- *RNA sequence / primary structure* $R[0..n) \in \Sigma^n$ $\quad \Sigma = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{U}\}$
- *RNA secondary structure*: matching of indices $S \subset [0..n)^2$ of pairs (i, j) that are
  - ordered $i \leq j$
  - disjoint: $(i, j), (k, l) \in S \wedge (i = k \vee j = l) \implies (i, j) = (k, l)$
  - not too close $(i, j) \in S \implies j - i \geq 4$ backbone can't bend more

    min. length of hairpin loop

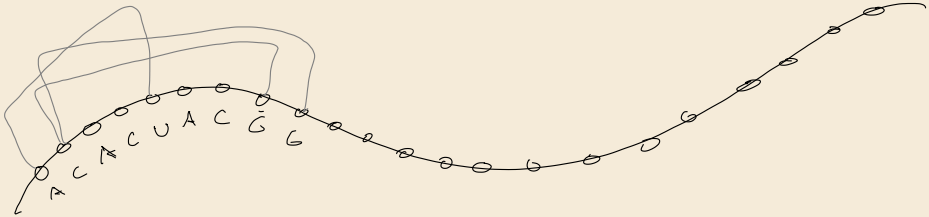


Model of RNA Structure

- *RNA sequence / primary structure* $R[0..n) \in \Sigma^n$ $\quad \Sigma = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{U}\}$
- *RNA secondary structure*: matching of indices $S \subset [0..n)^2$ of pairs (i, j) that are
  - ordered $i \leq j$
  - disjoint: $(i, j), (k, l) \in S \land (i = k \lor j = l) \implies (i, j) = (k, l)$
  - not too close $(i, j) \in S \implies j - i \geq 4$ backbone can't bend more
    min. length of hairpin loop
- secondary structure S is valid for sequence R if
  $(i, j) \in S \implies (R[i], R[j]) \in \mathcal{C} = \{(\mathtt{A}, \mathtt{U}), (\mathtt{U}, \mathtt{A}), (\mathtt{C}, \mathtt{G}), (\mathtt{G}, \mathtt{C}), (\mathtt{G}, \mathtt{U}), (\mathtt{U}, \mathtt{G})\}$
- $\mathcal{C}$ are the *canonical base pairs*: can form *hydrogen bonds* to stabilize RNA

Optimal RNA Structure – Attempt 1

- Since base pairs provide stability
  Try to maximize $|S|$ (# pairs) among all valid secondary structures for $R[0..n)$.



Optimal RNA Structure – Attempt 1

- Since base pairs provide stability
  Try to maximize $|S|$ (# pairs) among all valid secondary structures for $R[0..n)$.

$\rightsquigarrow$ **maximum matching** in graph of all bases

- possible in polynomial time
  - actually, ignoring minimum hairpin length, trivial greedy approach is optimal:
  1. form arbitrary `C` – `G` pairs (until we run out of `C`s or `G`s)
  2. form arbitrary `A` – `U` pairs (until we run out)
  3. form arbitrary `G` – `U` pairs (until we run out)

Optimal RNA Structure – Attempt 1

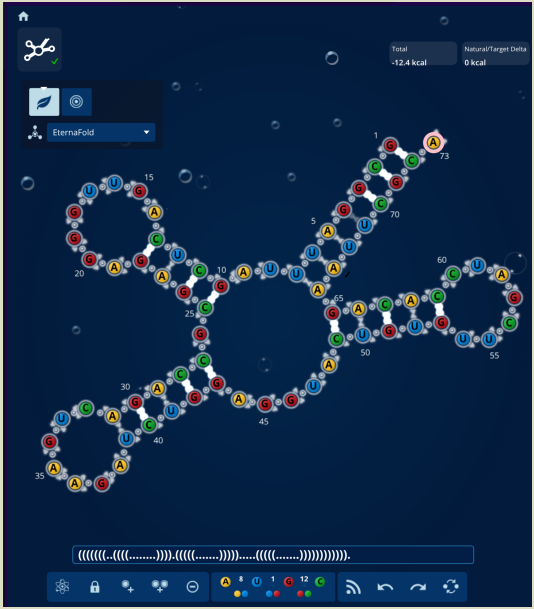
- Since base pairs provide stability
  Try to maximize $|S|$ (# pairs) among all valid secondary structures for $R[0..n)$.

$\rightsquigarrow$ **maximum matching** in graph of all bases

- possible in polynomial time
  - actually, ignoring minimum hairpin length, trivial greedy approach is optimal:
  1. form arbitrary `C` – `G` pairs (until we run out of `C`s or `G`s)
  2. form arbitrary `A` – `U` pairs (until we run out)
  3. form arbitrary `G` – `U` pairs (until we run out)

- unfortunately, useless predictions!
  - number of pairs dictated by base counts
  - many equally good options exist
  - many "optimal" solutions force entire molecule crowd up in one place

Let's play a game!



phenylalanine transfer RNA from Saccharomyces
https://rnacentral.org/rna/URS000011107D/4930

EteRNA

`eternagame.org`

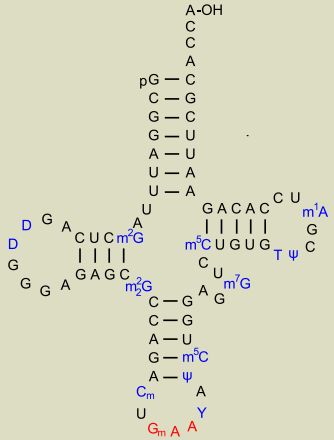
- Eterna is a citizen scientist computer game running since 2010
  lead by Rhiju Das (Stanford University School of Medicine)
- You have to design an RNA sequence that folds into a given *target secondary structure*.
- The game uses the best available simulation of RNA folding.
- Simulation, prediction, and RNA design algorithms are co-evolving
  - RNA design crowdsourced to players
  - top designs synthesized and structure determined

  ⇝ growing dataset for RNA structures

2D Approximation

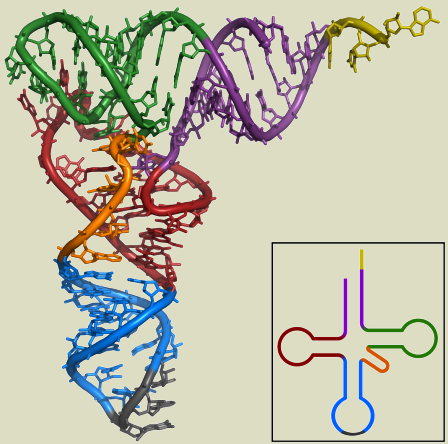
- As in Eterna, RNA secondary structure often drawn as *"roadkill diagrams"*

Roadkill diagram of yeast Phe tRNA



https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_blanco.svg

3D Structure of yeast Phe tRNA



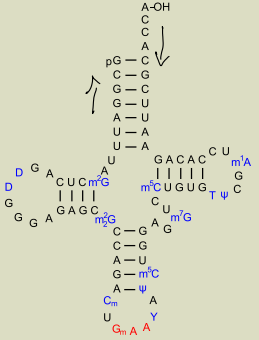
https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_1ehz.png

Stacks

Key Observation: Stable structures have many **adjacent pairs**

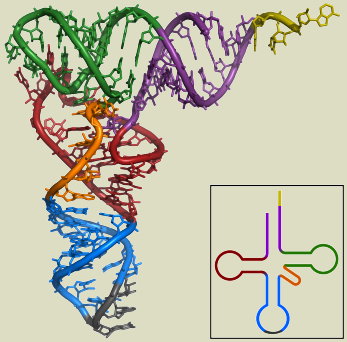
- *"stacked" pairs* forming a *stem* (the "ladder" regions)
- in 3D, stems form into a double helix (similar to DNA!)
- only reverse complement stems are stable

Roadkill diagram of yeast Phe tRNA



https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_blanco.svg

3D Structure of yeast Phe tRNA

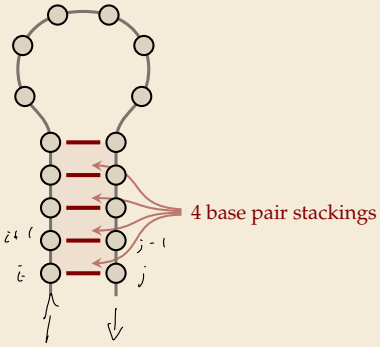


https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_1ehz.png

Optimal RNA Structure – Attempt 2

- Recall: $S \subset [0..n)^2$ set of indices of paired bases
- instead of maximizing $|S|$ (# pairs), let's maximize number of **base pair stackings**!

$$\mathrm{BPS}(S) = \left|\left\{(i,j) \in S : (i+1, j-1) \in S\right\}\right|$$



General Secondary Structure Prediction

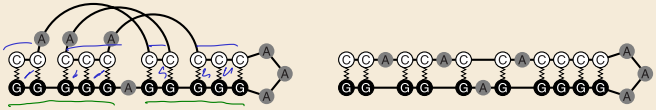
- **Given:** Sequence $R \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{U}\}^n$
- **Goal:** *Valid* secondary structure S with maximal $\mathrm{BPS}(S)$

Hardness

Unfortunately, General Secondary Structure Prediction is NP-hard.

- reduction from BinPacking

Lyngsø: *Complexity of Pseudoknot Prediction in Simple Models*, ICALP **2004**



(a) An optimum structure for the RNA sequence constructed from an instance of Bin Packing with four items of sizes 2, 2, 3, and 3, and two bins of capacity 5.

(b) An optimum structure for the RNA sequence constructed from an instance of Bin Packing with four items of sizes 2, 2, 2, and 4, and two bins of capacity 5.

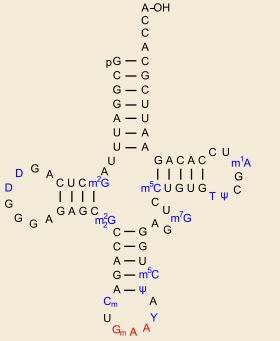
Fig. 3. Illustration of how the number of helices can be kept to one per item for an RNA sequence constructed from a 'yes' instance of Bin Packing, while the base pairs of at least one substring corresponding to an item have to be split over at least two helices if the RNA sequence is constructed from a 'no' instance of Bin Packing.

8.3 Pseudoknot-free secondary structures

Flat Structures

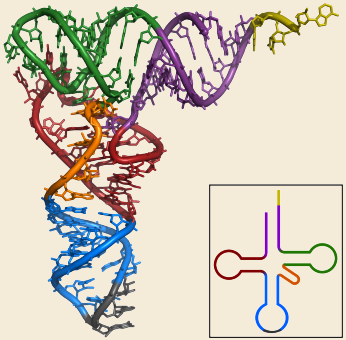
Recall example tRNA structure

Roadkill diagram of yeast Phe tRNA



https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_blanco.svg

3D Structure of yeast Phe tRNA

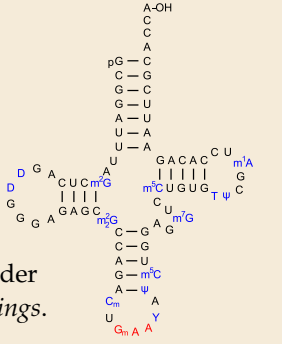


https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_1ehz.png

Flat Structures

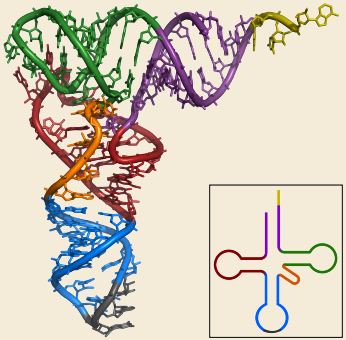
Recall example tRNA structure

Roadkill diagram of yeast Phe tRNA



https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_blanco.svg

3D Structure of yeast Phe tRNA

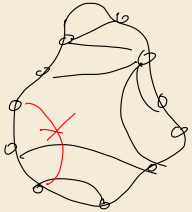


https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_1ehz.png

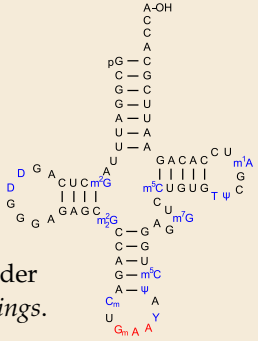
⇝ Seems reasonable to only consider roadkill diagrams *without crossings*.

Flat Structures

Recall example tRNA structure

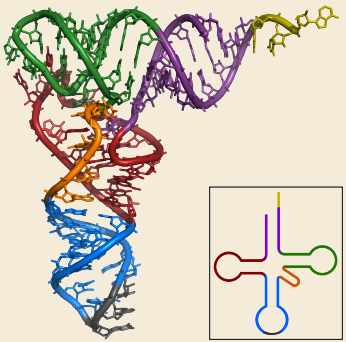


Roadkill diagram of yeast Phe tRNA



https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_blanco.svg

3D Structure of yeast Phe tRNA



https://commons.wikimedia.org/wiki/File:TRNA-Phe_yeast_1ehz.png

⇝ Seems reasonable to only consider roadkill diagrams *without crossings*.

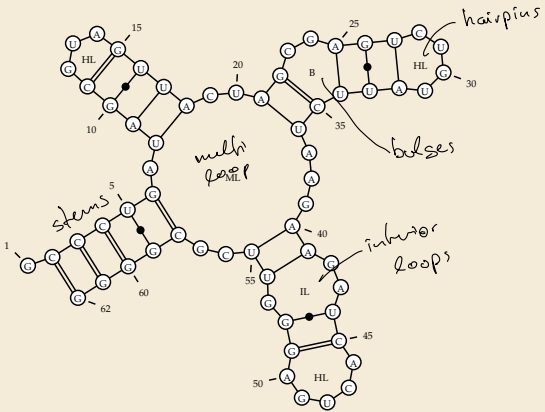
"Correct" formalization seems to be:
Require graph of pairs bases and backbone edges to be ***outerplanar***.

Any other secondary structure is called a *pseudoknot*.

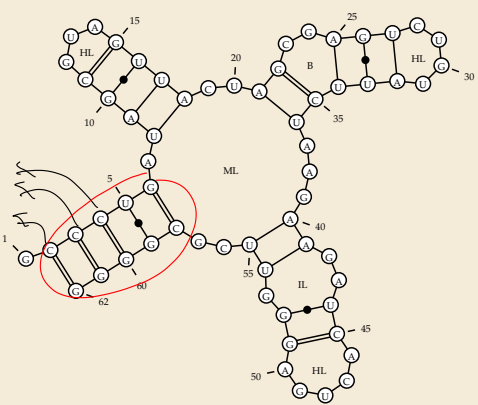
Pseudoknot-free secondary structures

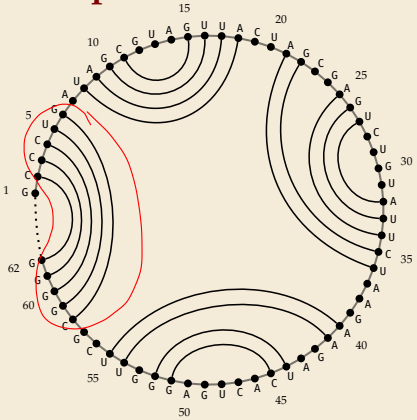
- planar secondary structure (pairs) cover most of *free energy* of folding
- "coarse graining" of 3D structure biochemically useful
- natural intermediate step on folding pathway
- often well conserved between related species
- computationally tractable

Pseudoknot-free secondary structures – Representations

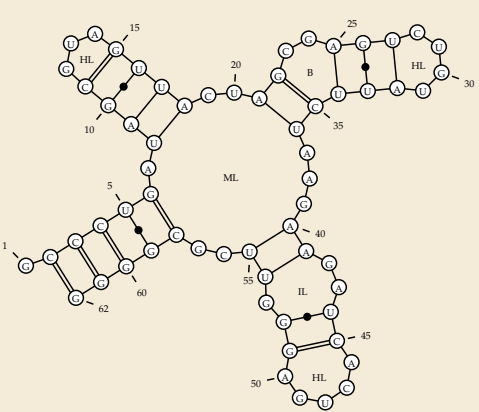


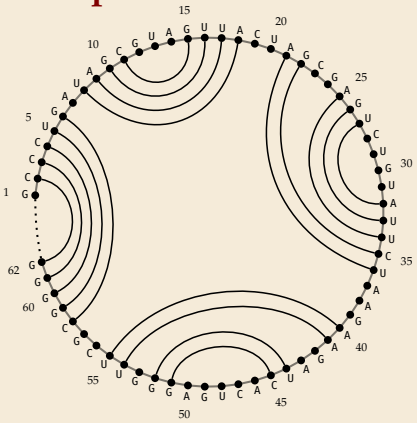
Pseudoknot-free secondary structures – Representations

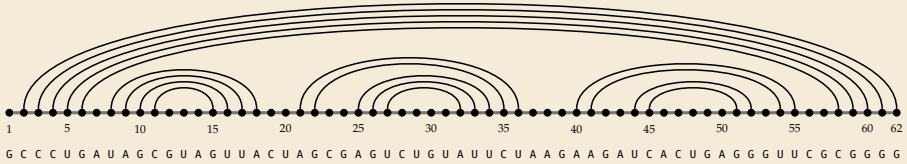




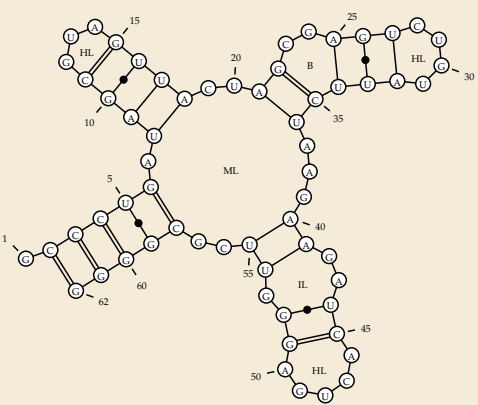
Pseudoknot-free secondary structures – Representations

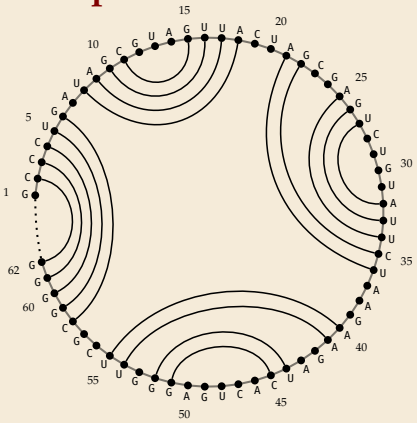


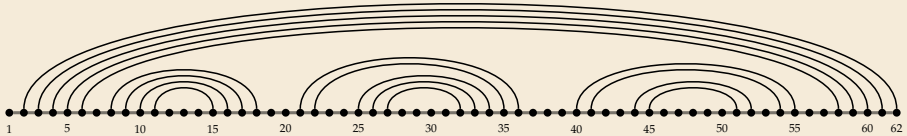




Pseudoknot-free secondary structures – Representations

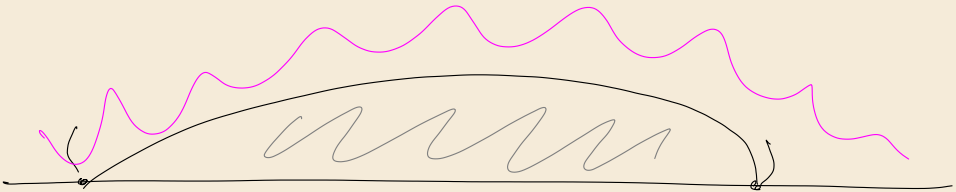






Nussinov's Algorithm

Idea: *Maximize total number of valid pairs among all* ***pseudoknot-free*** *structures.*



Nussinov's Algorithm

Idea: *Maximize total number of valid pairs among all* ***pseudoknot-free*** *structures.*

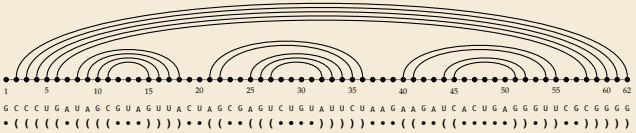
- back to maximum matching, but subject to outerplanar constraint . . .

Nussinov's Algorithm

Idea: *Maximize total number of valid pairs among all* ***pseudoknot-free*** *structures.*

- back to maximum matching, but subject to outerplanar constraint . . .
- key insight: *decomposability!*

see arc diagram / dot-bracket representation

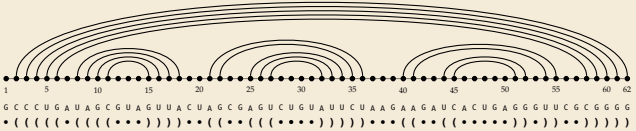


Nussinov's Algorithm

Idea: *Maximize total number of valid pairs among all* ***pseudoknot-free*** *structures.*

- back to maximum matching, but subject to outerplanar constraint . . .
- key insight: *decomposability!*

see arc diagram / dot-bracket representation



⇝ Apply dynamic programming on subproblems $R[i..j]$

$D(i, j)$ = max valid pairs in pseudoknot-free structure for $R[i..j]$

Nussinov's Algorithm – DP

$D(i,j)$ = max valid pairs in any pseudoknot-free structure for $R[i..j]$

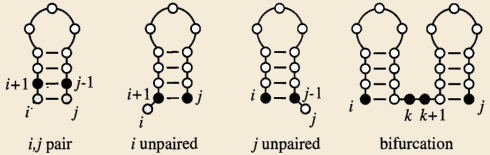
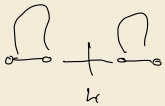


Figure 10.7 from Durbin et al. 1998



$$\rightsquigarrow \quad D(i,j) \;=\; \begin{cases} 0, & \text{if } j-i \le 4; \\ \max \begin{cases} D(i+1,j-1) + \left[(R[i],R[j-1]) \in \mathcal{C}\right], \\ D(i+1,j), \\ D(i,j-1), \\ \max\limits_{k \in [i..j)} D(i,k) + D(k+1,j) \end{cases} & \text{else.} \end{cases}$$

$\rightsquigarrow$ $O(n^3)$ time, $O(n^2)$ space

8.5 Refined Models

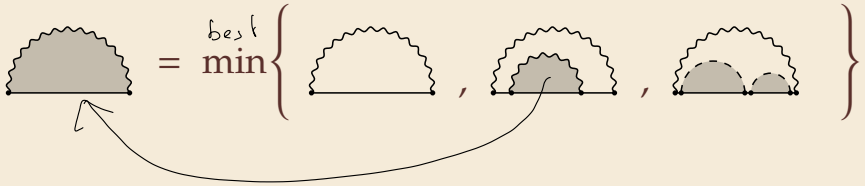
Back to Base Pair Stackings

- While maximum outerplanar matching is well-defined and tractable, it doesn't usually yield natural structures.
- already know that we should count base pair stackings!

Back to Base Pair Stackings

- While maximum outerplanar matching is well-defined and tractable, it doesn't usually yield natural structures.
- already know that we should count base pair stackings!
- We can extend the DP solution to count those instead!

Graphical notation for DP recursions

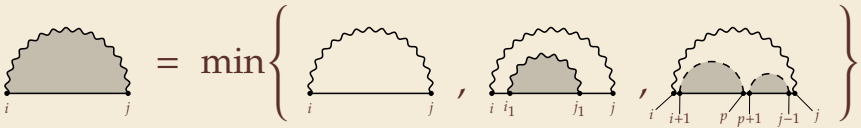


Key

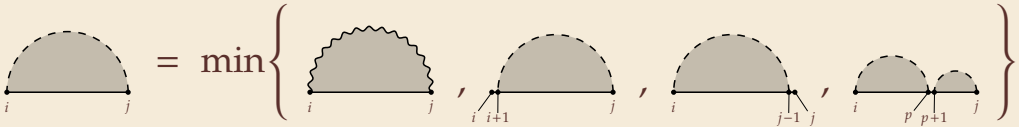
- dots — bases; if touching, neighbors on backbone
- horizontal line — RNA backbone
- wiggly arcs — base pair
- dashed arcs — boundary; could be paired or not
- white area — no arcs here
- gray area — potentially further arcs

Counting Base Pair Stackings

Idea: Need to remember whether outermost bases paired.

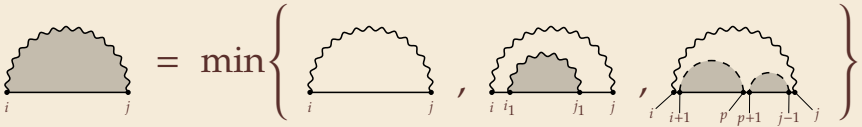


- In the middle case, if $(i_1, j_1) = (i, j)$, count stacked base pair for (i, j)

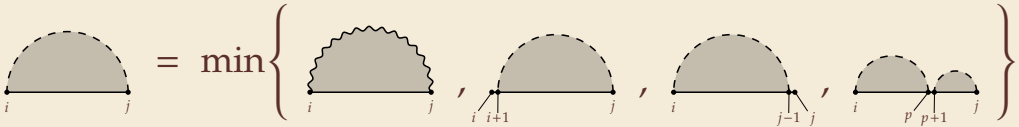


Counting Base Pair Stackings

Idea: Need to remember whether outermost bases paired.



- In the middle case, if $(i_1, j_1) = (i, j)$, count stacked base pair for (i, j)



$\rightsquigarrow$ Same $O(n^3)$ time, $O(n^2)$ space complexity

Turner Energy Model

- Simply counting base pair stackings is still a **very crude approximation**
- Which bases are paired influences bonding strength
- Which bases are adjacent in stems influences stabilization contribution of stem
- Which bases form first unpaired base in hairpin loop influences stability
- . . . (play Eterna a bit for more 😉)

Turner Energy Model

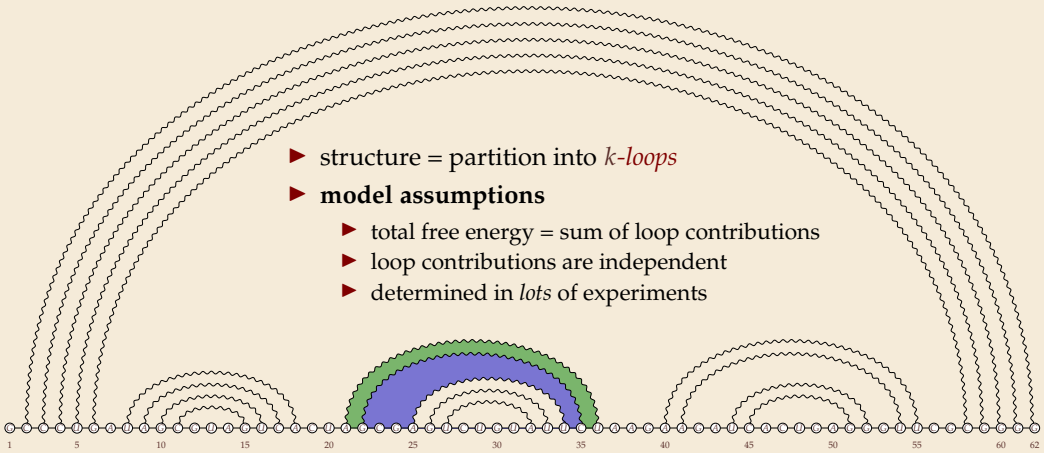
- Simply counting base pair stackings is still a **very crude approximation**
- Which bases are paired influences bonding strength
- Which bases are adjacent in stems influences stabilization contribution of stem
- Which bases form first unpaired base in hairpin loop influences stability
- . . . (play Eterna a bit for more 😉)

⇝ More refined models to compute free energy (≈ instability) of structure

Mathews, Sabina, Zuker, Turner: *Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure,* J Molecul. Biolog. **1999**

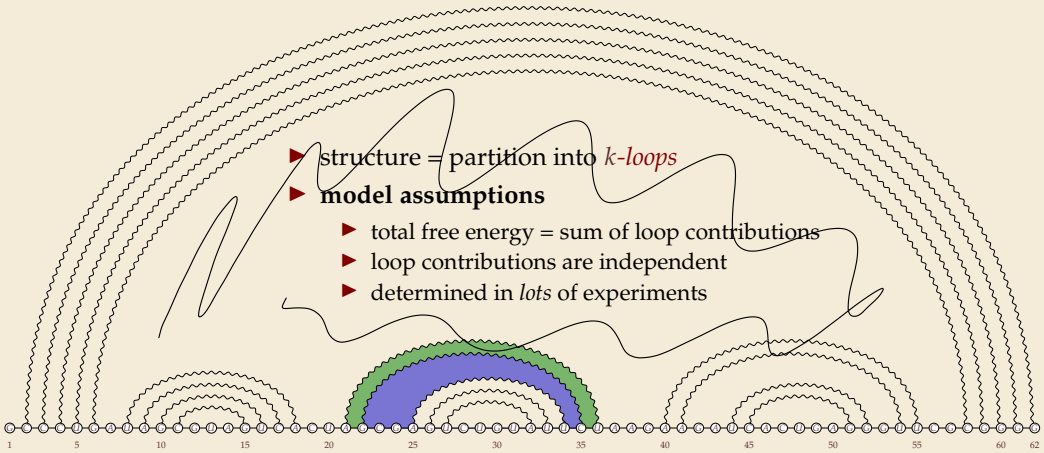
Mathews, Disney, Childs, Schroeder, Zuker, Turner: *Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure,* PNAS **2004**

Turner Energy Model [2]



- structure = partition into *k-loops*
- **model assumptions**
  - total free energy = sum of loop contributions
  - loop contributions are independent
  - determined in *lots* of experiments

Turner Energy Model [2]



- structure = partition into *k-loops*
- **model assumptions**
  - total free energy = sum of loop contributions
  - loop contributions are independent
  - determined in *lots* of experiments

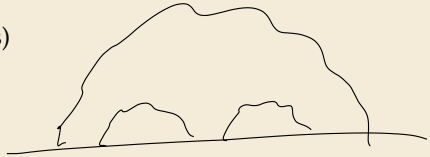
Conceptually unbounded sum

too many variables!

Zuker's Algorithm

- Only compute exactly up to 2-loops (2 enclosed pairs)
- additive approximation for bigger multiloops

⇝ *same* mutually recursive cost as for pair stackings



8.6 Grammar-based Approaches

Can’t machine learning help?

- free-energy models are great *ab initio* methods
- however, they remain limited in accuracy
- with growing datasets, tempting to improve structure prediction using machine learning

Can’t machine learning help?

- free-energy models are great *ab initio* methods
- however, they remain limited in accuracy
- with growing datasets, tempting to improve structure prediction using machine learning

- but: available data much too few for blackbox learning

⇝ statistical learning with curated probabilistic model

Probabilistic Context-Free Grammars

Recap from your formal languages intro course . . .

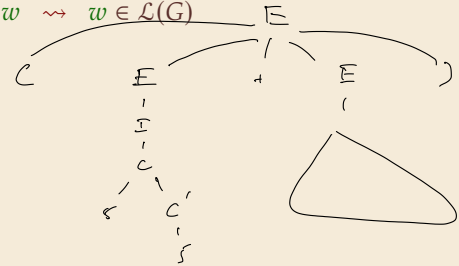
Context-free grammars (CFG)

$G = (N, T, R, S)$

- nonterminals N
- terminals T
- rules $R \subseteq N \times (N \cup T)^*$
- start symbol $S \in N$

Applying rules to replace nonterminals

$S \Rightarrow^* w \quad \rightsquigarrow \quad w \in \mathcal{L}(G)$



Example

- $N = \{E, I, V, C, C'\}$
- $T = \{\texttt{x}, \texttt{y}, \texttt{0}, \ldots, \texttt{9}, \texttt{+}, \bullet, (,)\}$
- $E \to (E + E) \mid (E \bullet E) \mid I$
  $I \to C \mid V$
  $V \to \texttt{x} \mid \texttt{y}$
  $C \to \texttt{0} \mid \texttt{1}C' \mid \ldots \mid \texttt{9}C'$
  $C' \to \varepsilon \mid \texttt{0}C' \mid \ldots \mid \texttt{9}C'$

  empty string

(55 + (5.0))

Probabilistic Context-Free Grammars

Recap from your formal languages intro course . . .

Context-free grammars (CFG)

$G = (N, T, R, S)$

- nonterminals N
- terminals T
- rules $R \subseteq N \times (N \cup T)^*$
- start symbol $S \in N$

Applying rules to replace nonterminals

$S \Rightarrow^* w \quad \rightsquigarrow \quad w \in \mathcal{L}(G)$

Example

- $N = \{E, I, V, C, C'\}$
- $T = \{\texttt{x}, \texttt{y}, \texttt{0}, \ldots, \texttt{9}, \texttt{+}, \bullet\}$
- $E \to (E + E) \mid (E \bullet E) \mid I$
  $I \to C \mid V$
  $V \to \texttt{x} \mid \texttt{y}$
  $C \to \texttt{0} \mid \texttt{1}C' \mid \ldots \mid \texttt{9}C'$
  $C' \to \varepsilon \mid \texttt{0}C' \mid \ldots \mid \texttt{9}C'$

  ε: empty string

Probabilistic Context-Free Grammars (PCFG)

generalization of Markov chain

- For each nonterminal, assign *probabilities* to right-hand sides.

$\rightsquigarrow$ prob of a derivation in G = product of rule probabilities.