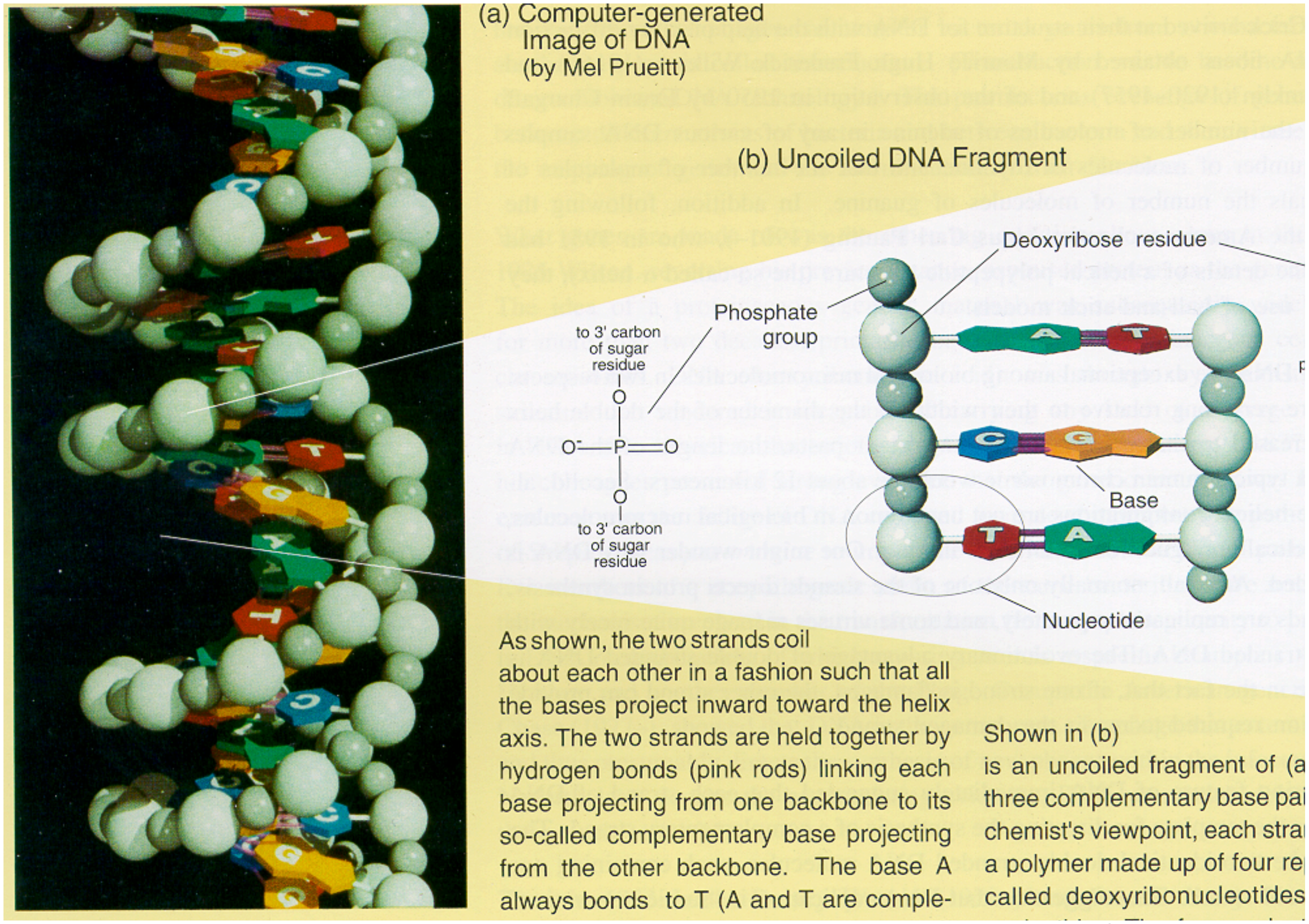
CSE 417: Algorithms and Computational Complexity

Winter 2009

W. L. Ruzzo

Dynamic Programming, II
RNA Folding

The Double Helix


(a) Computer-generated Image of DNA (by Mel Prueitt)
(b) Uncoiled DNA Fragment
Deoxyribose residue
Phosphate group
to 3' carbon of sugar residue
O
O⁻—P═O
O
to 3' carbon of sugar residue
A
T
C
G
Base
T
A
Nucleotide

As shown, the two strands coil about each other in a fashion such that all the bases project inward toward the helix axis. The two strands are held together by hydrogen bonds (pink rods) linking each base projecting from one backbone to its so-called complementary base projecting from the other backbone. The base A always bonds to T (A and T are comple-

Shown in (b) is an uncoiled fragment of (a three complementary base pai chemist's viewpoint, each strai a polymer made up of four re called deoxyribonucleotides

Los Alamos Science

http://www.rcsb.org/pdb/explore.do?structureId=1GAT

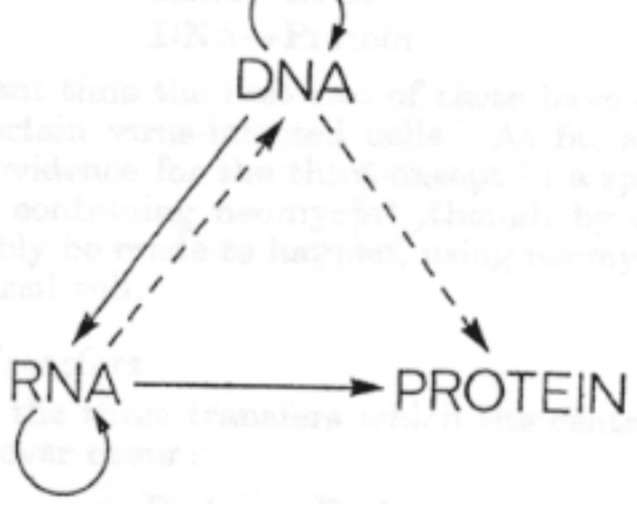
NATURE VOL. 227 AUGUST 8 1970

Central Dogma of Molecular Biology

by
FRANCIS CRICK
MRC Laboratory
Hills Road,
Cambridge CB2 2QH

The central dogma of molecular biology deals with the detailed residue-by-residue transfer of sequential information. It states that such information cannot be transferred from protein to either protein or nucleic acid.

"The central dogma, enunciated by Crick in 1958 and the keystone of molecular biology ever since, is likely to prove a considerable over-simplification."

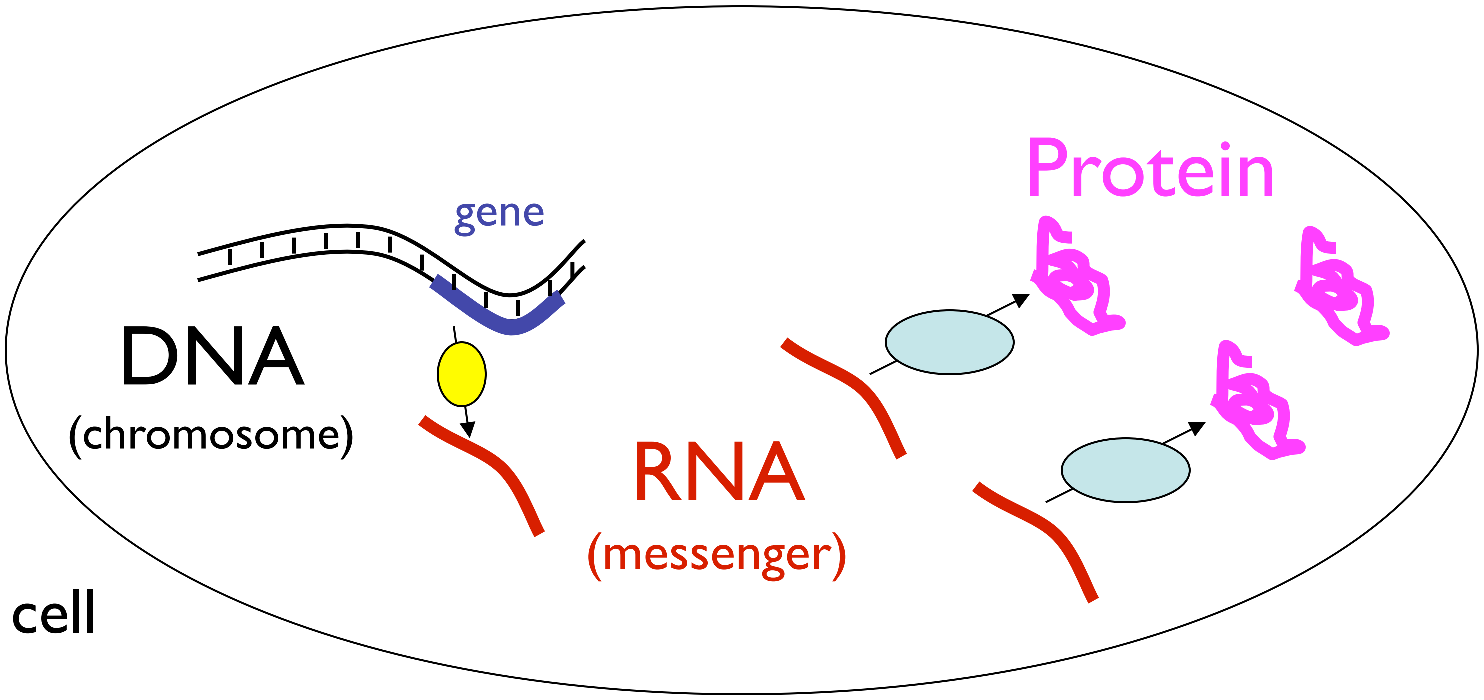


$$\begin{tikzcd}
& \text{DNA} \arrow[loop above] \arrow[dl, shift right] \arrow[dr, dashed] & \\
\text{RNA} \arrow[loop below] \arrow[ur, dashed, shift right] \arrow[rr] & & \text{PROTEIN}
\end{tikzcd}$$

Fig. 2. The arrows show the situation as it seemed in 1958. Solid arrows represent probable transfers, dotted arrows possible transfers. The absent arrows (compare Fig. 1) represent the impossible transfers postulated by the central dogma. They are the three possible arrows starting from protein.

The "Central Dogma" of Molecular Biology

DNA $\longrightarrow$ RNA $\longrightarrow$ Protein



Non-coding RNA

Messenger RNA - codes for proteins

Non-coding RNA - all the rest

- Before, say, mid 1990's, 1-2 dozen known
  (critically important, but narrow roles)

Since mid 90's dramatic discoveries

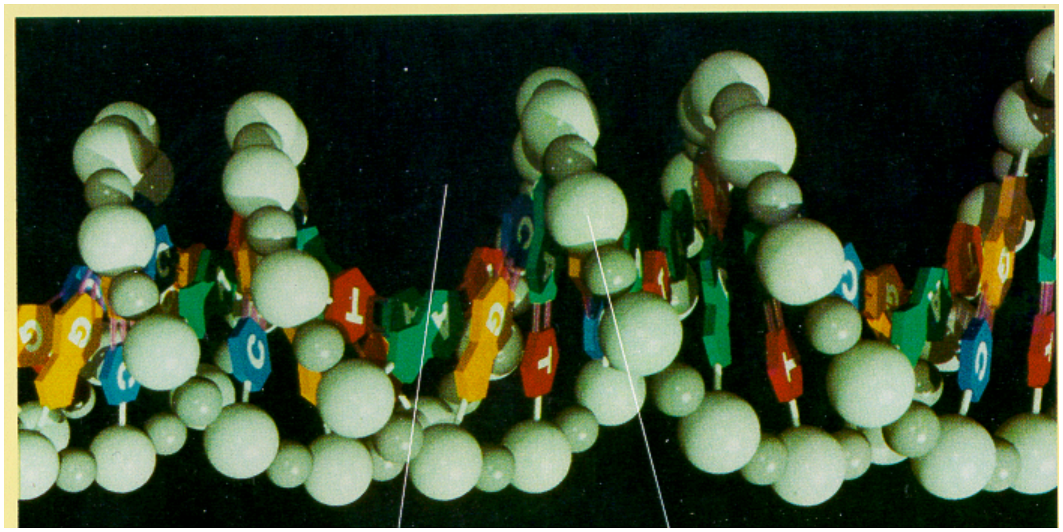
- Regulation, transport, stability/degradation
- E.g. "microRNA": 100s in humans => 50% of genes
- E.g. "riboswitches": 1000s in bacteria

DNA structure: dull

5'…ACCGCTAGATG…3'

| | | | | | | | | | |

3'…TGGCGATCTAC…5'

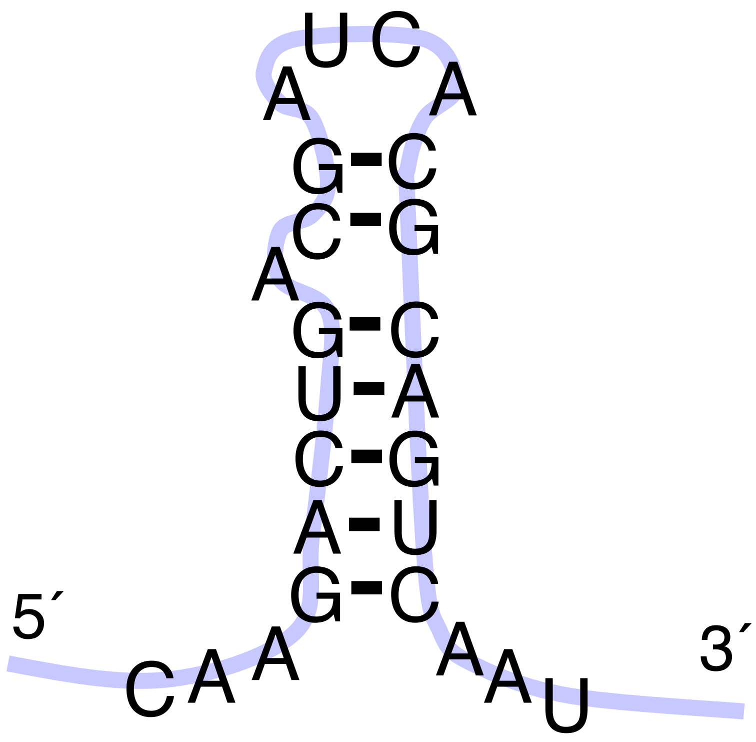


RNA Secondary Structure:
RNA makes helices too

Base pairs

A-U

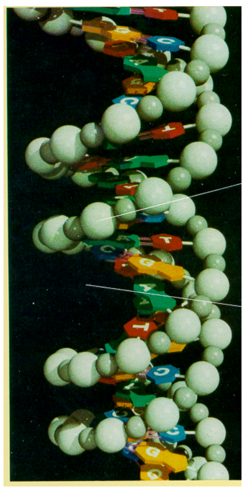
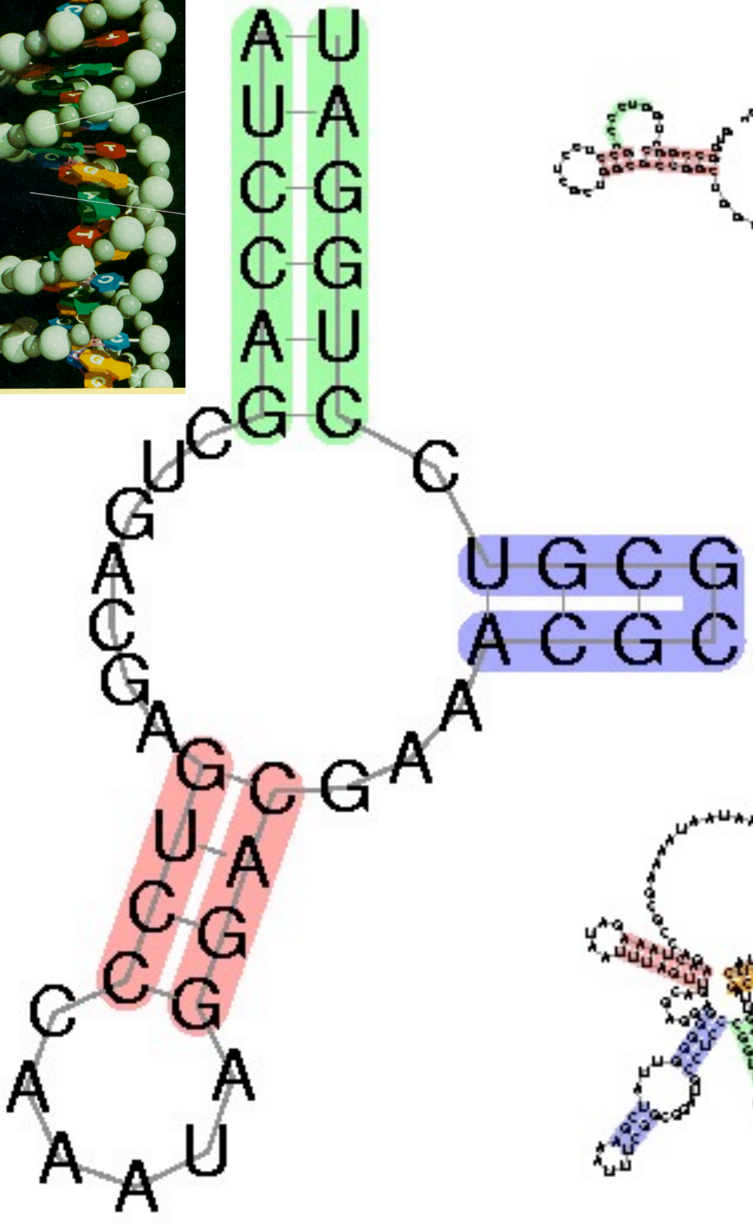
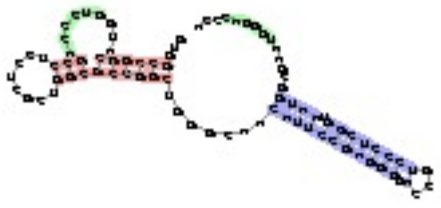
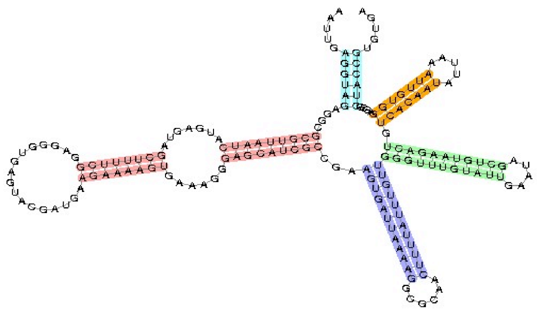
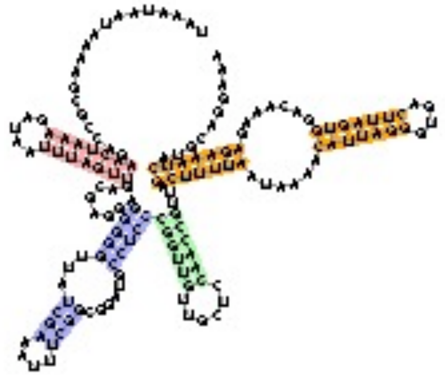
C-G



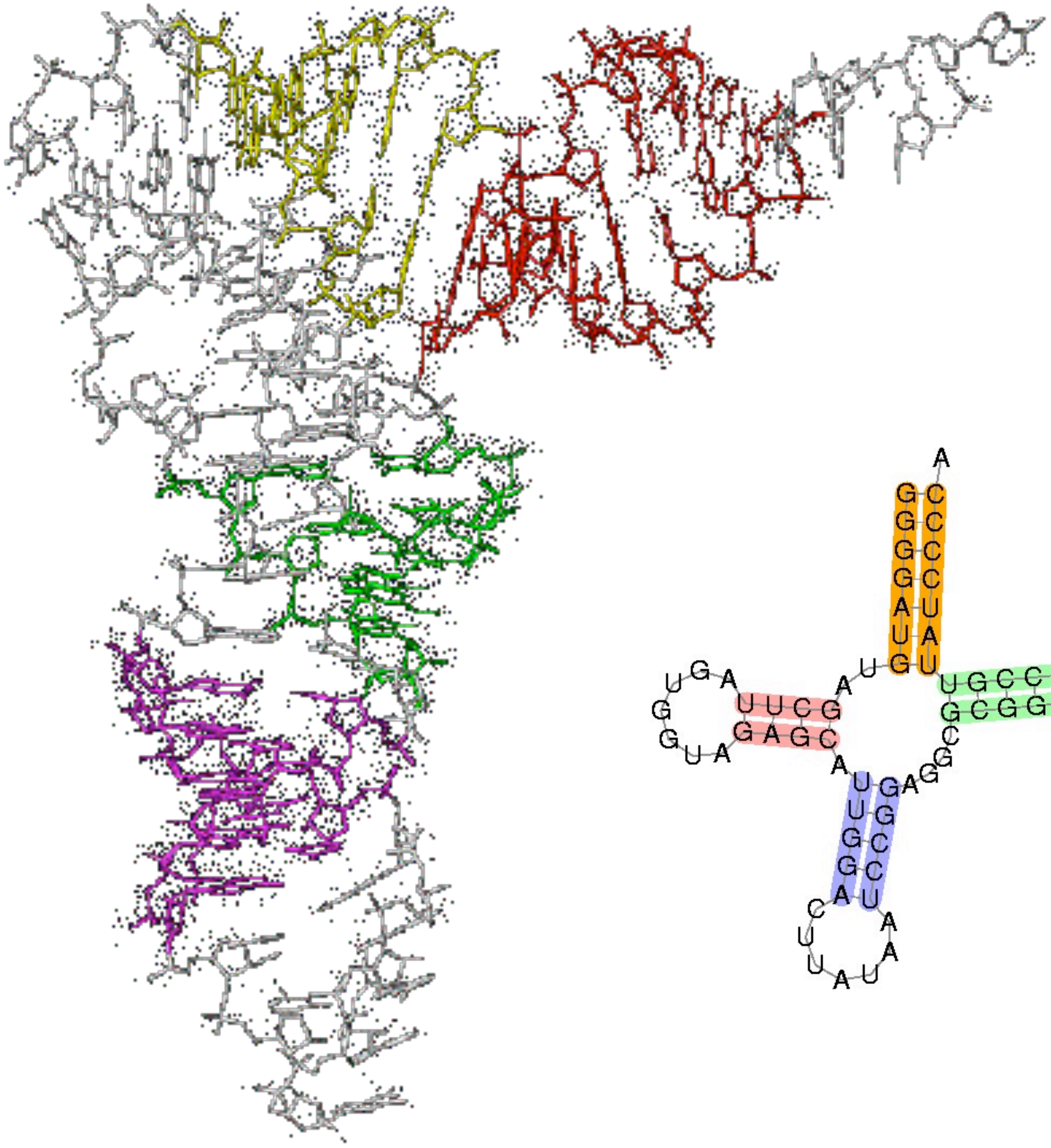
Usually *single* stranded

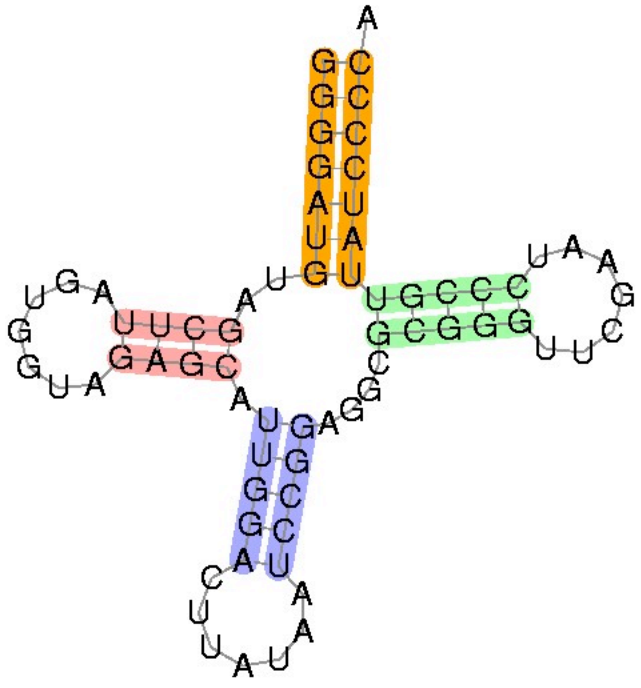
RNA Structure: Rich

- RNA's fold, and function
- Nature uses what works

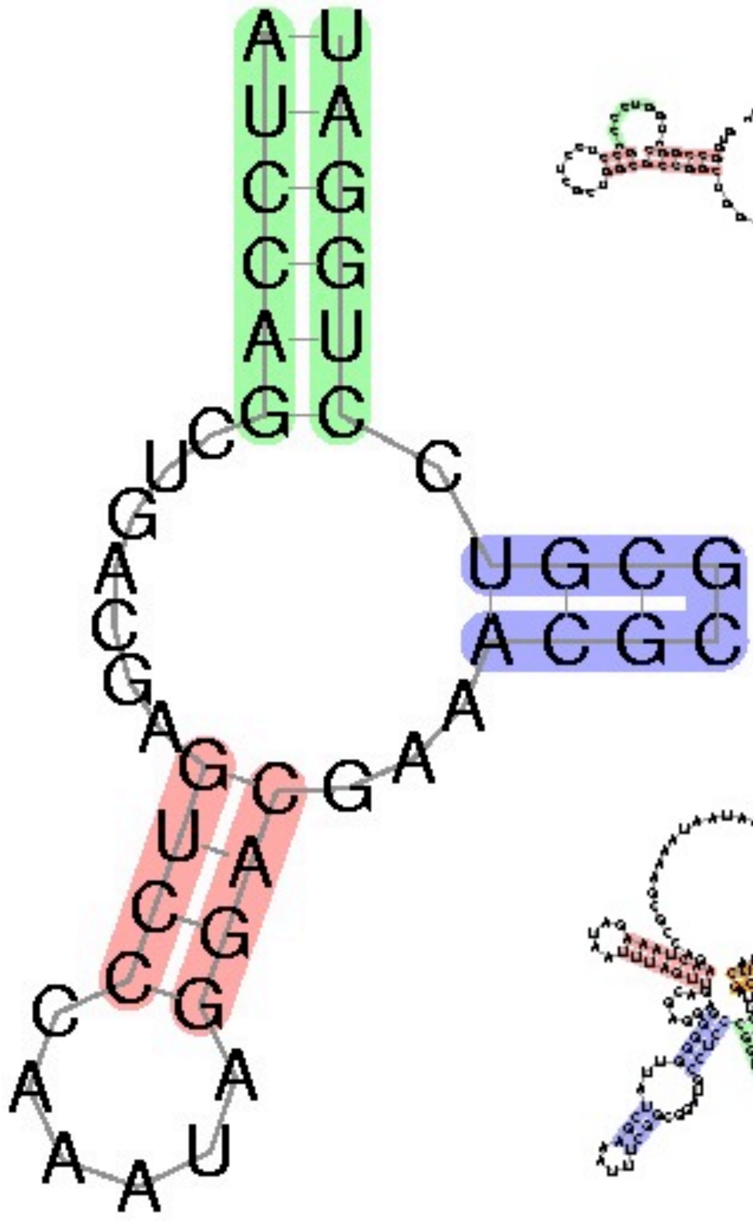
http://www.rcsb.org/pdb/explore.do?structureId=1EVV

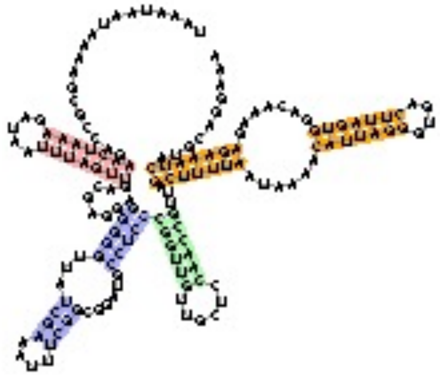


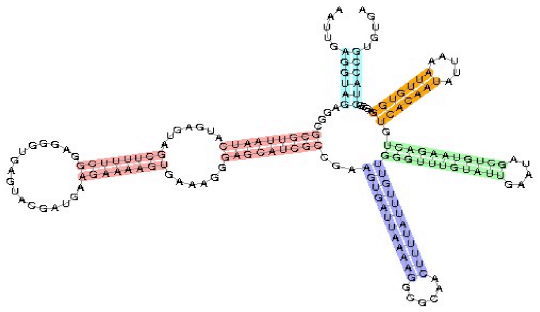


RNA Secondary Structure:

Not everything, but important, easier than 3d



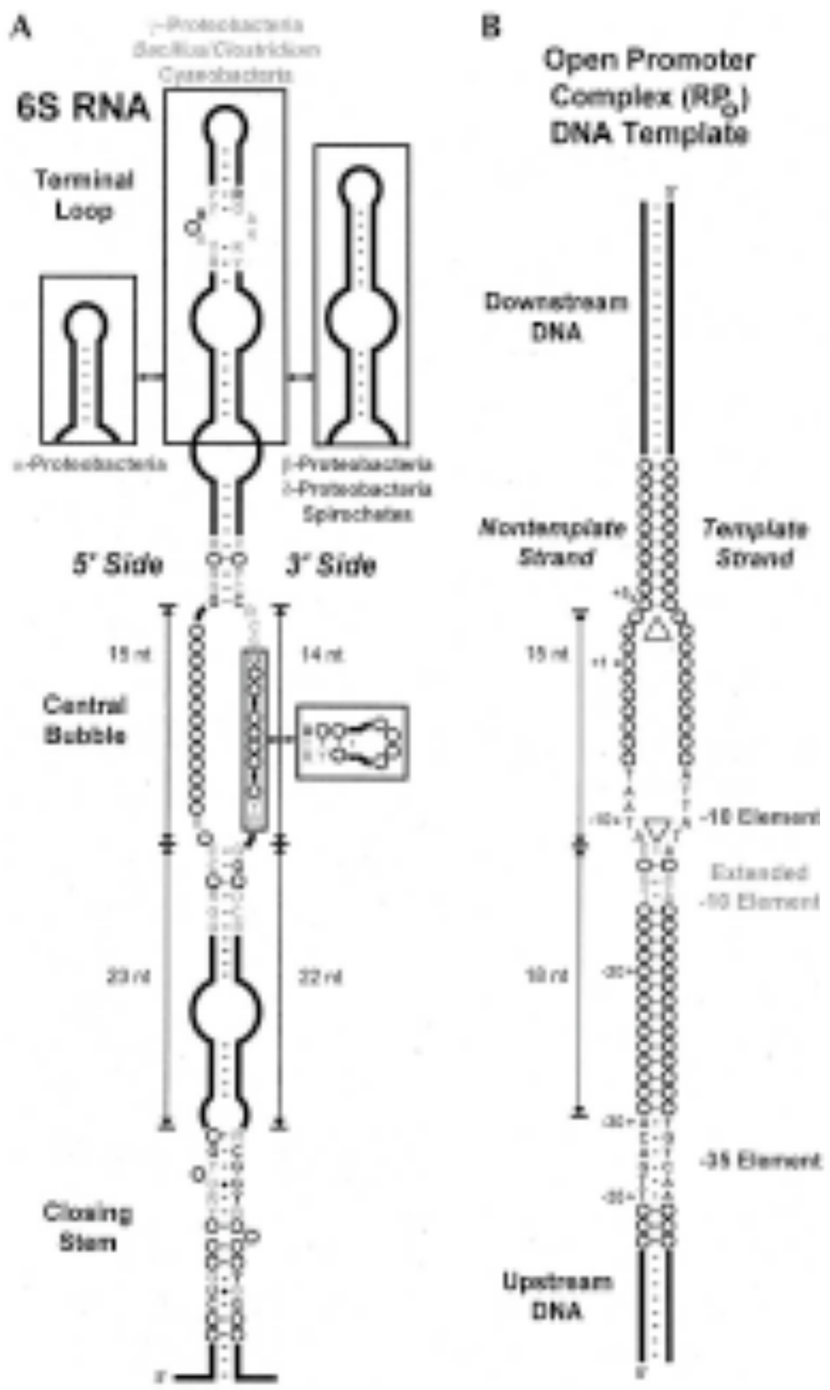


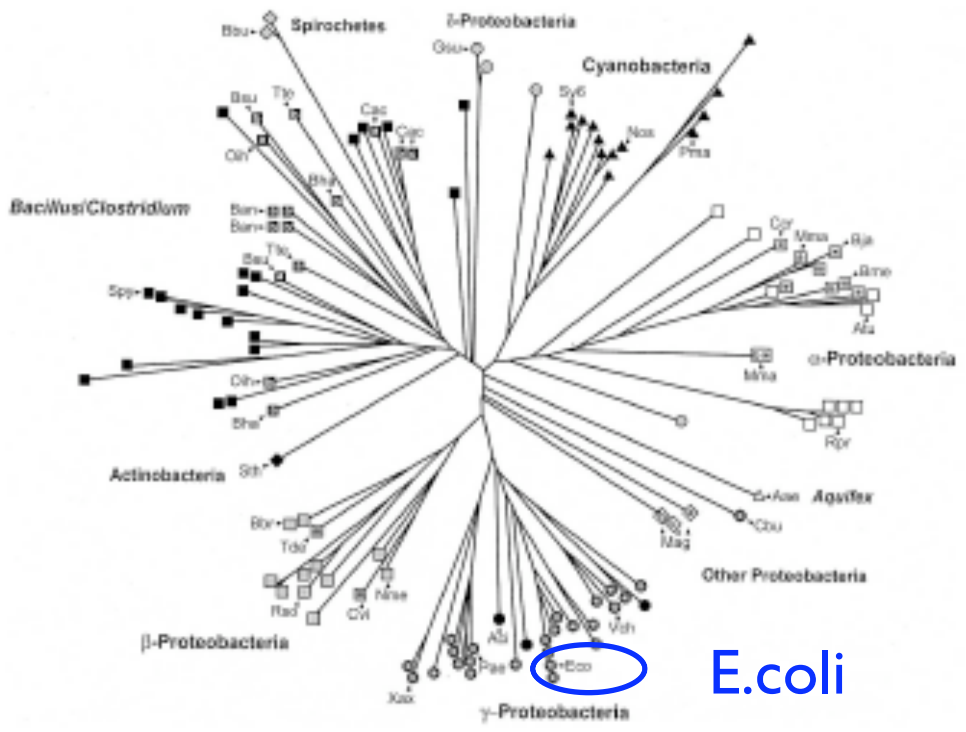


Why is structure important?

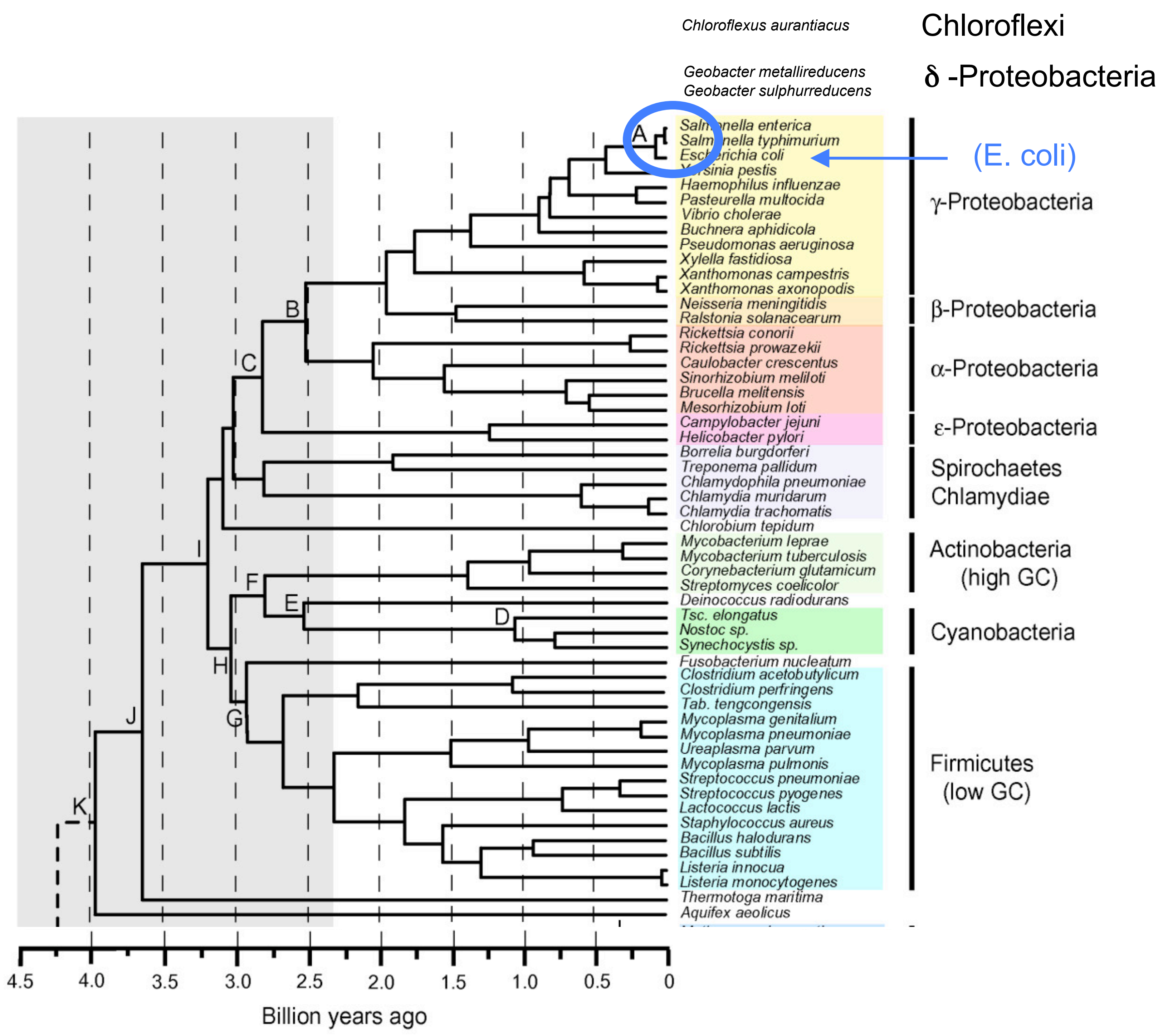
- For protein-coding, similarity in sequence is a powerful tool for finding related sequences
  - e.g. “hemoglobin” is easily recognized in all vertebrates
- For non-coding RNA, many different sequences have the same structure, and structure is most important for function.
  - So, using structure plus sequence, can find related sequences at much greater evolutionary distances

6S mimics an open promoter

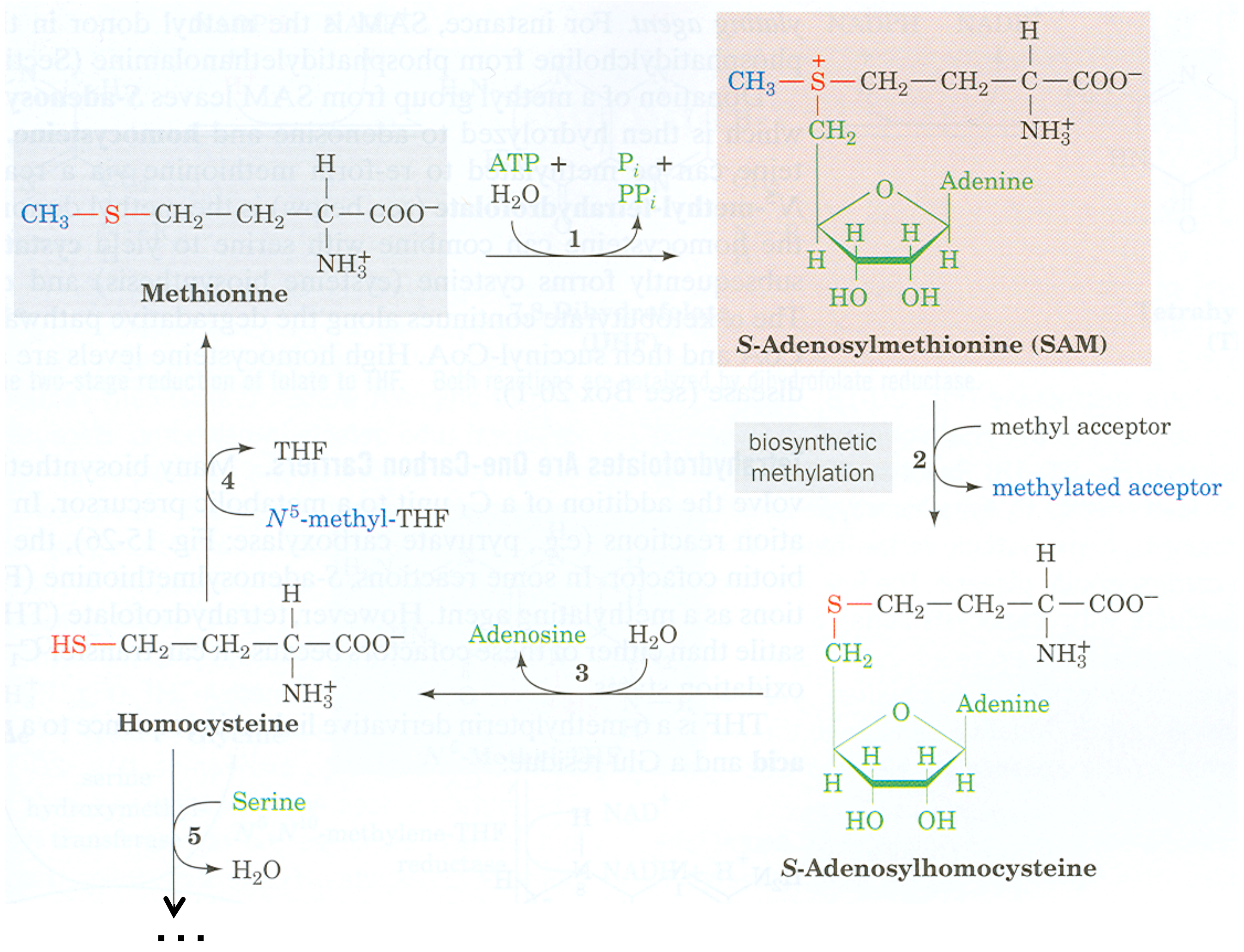




Barrick et al. *RNA* 2005
Trotochaud et al. *NSMB* 2005
Willkomm et al. NAR 2005

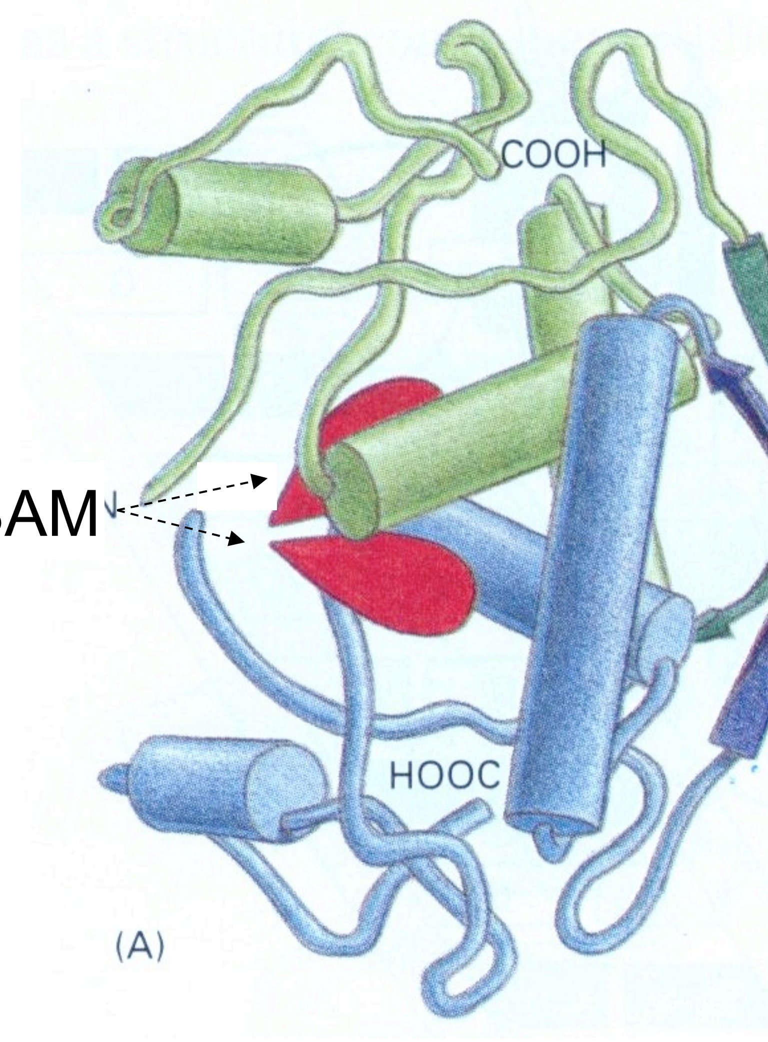

Chloroflexus aurantiacus
Chloroflexi
Geobacter metallireducens
Geobacter sulphurreducens
δ -Proteobacteria
(E. coli)
A
B
C
D
E
F
G
H
I
J
K
Salmonella enterica
Salmonella typhimurium
Escherichia coli
Yersinia pestis
Haemophilus influenzae
Pasteurella multocida
Vibrio cholerae
Buchnera aphidicola
Pseudomonas aeruginosa
Xylella fastidiosa
Xanthomonas campestris
Xanthomonas axonopodis
γ-Proteobacteria
Neisseria meningitidis
Ralstonia solanacearum
β-Proteobacteria
Rickettsia conorii
Rickettsia prowazekii
Caulobacter crescentus
Sinorhizobium meliloti
Brucella melitensis
Mesorhizobium loti
α-Proteobacteria
Campylobacter jejuni
Helicobacter pylori
ε-Proteobacteria
Borrelia burgdorferi
Treponema pallidum
Chlamydophila pneumoniae
Chlamydia muridarum
Chlamydia trachomatis
Spirochaetes
Chlamydiae
Chlorobium tepidum
Mycobacterium leprae
Mycobacterium tuberculosis
Corynebacterium glutamicum
Streptomyces coelicolor
Actinobacteria
(high GC)
Deinococcus radiodurans
Tsc. elongatus
Nostoc sp.
Synechocystis sp.
Cyanobacteria
Fusobacterium nucleatum
Clostridium acetobutylicum
Clostridium perfringens
Tab. tengcongensis
Mycoplasma genitalium
Mycoplasma pneumoniae
Ureaplasma parvum
Mycoplasma pulmonis
Streptococcus pneumoniae
Streptococcus pyogenes
Lactococcus lactis
Staphylococcus aureus
Bacillus halodurans
Bacillus subtilis
Listeria innocua
Listeria monocytogenes
Firmicutes
(low GC)
Thermotoga maritima
Aquifex aeolicus
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0
Billion years ago

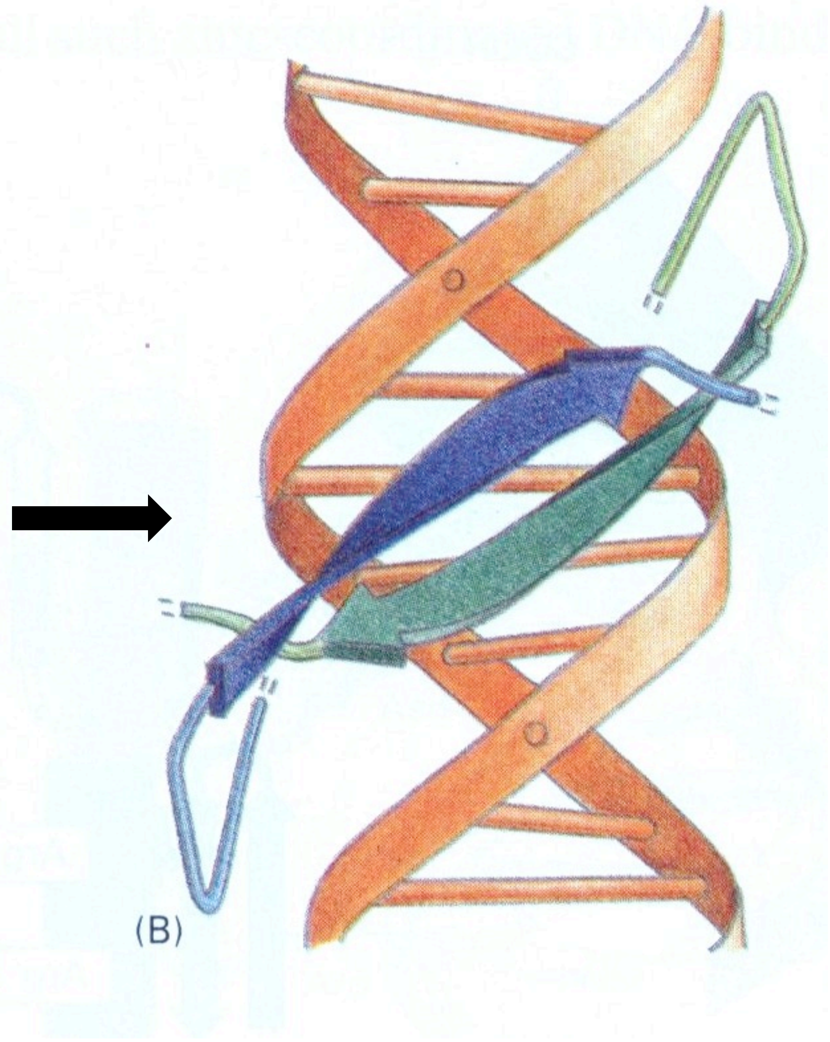
In Bacteria: A typical biosynthetic cycle around a critical metabolite ("SAM")



Gene Regulation: The MET Repressor

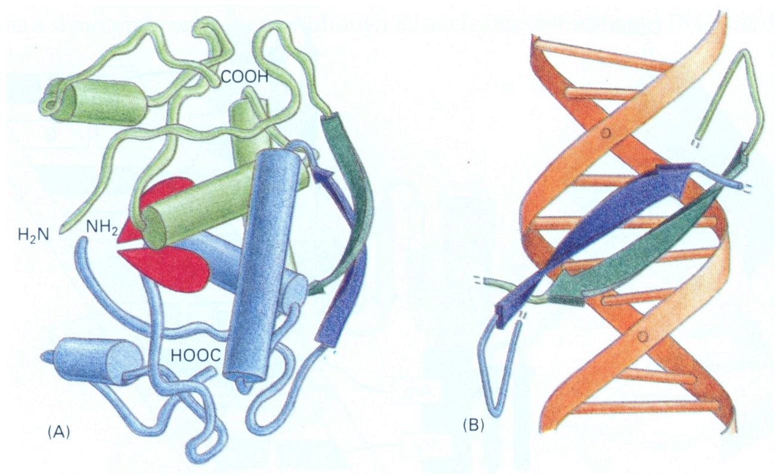
SAM





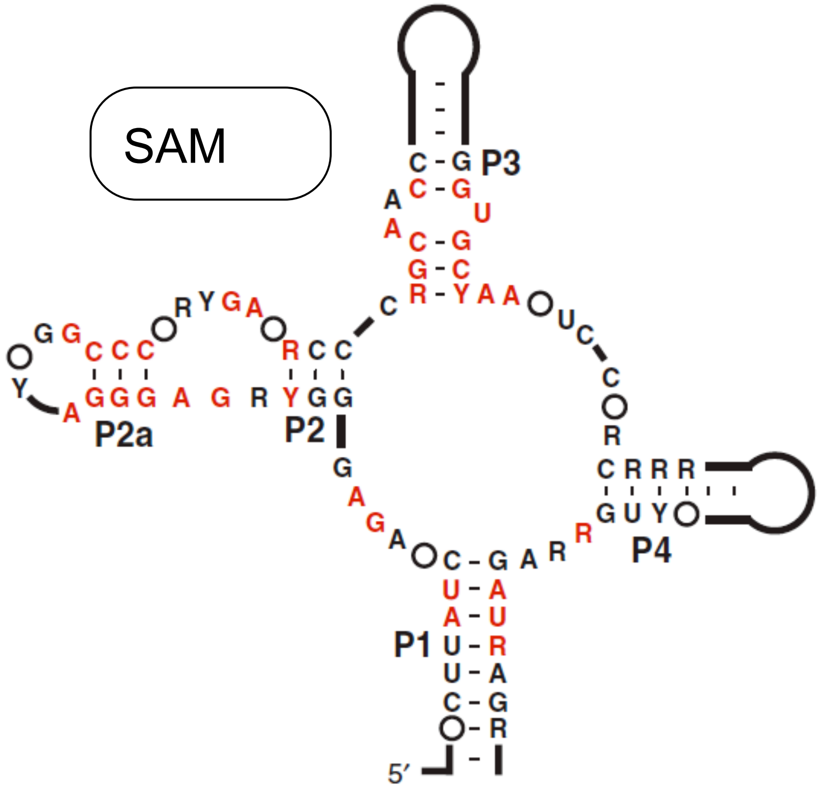
Protein Alberts, et al, 3e. DNA

Alberts, et al, 3e.



← The protein way

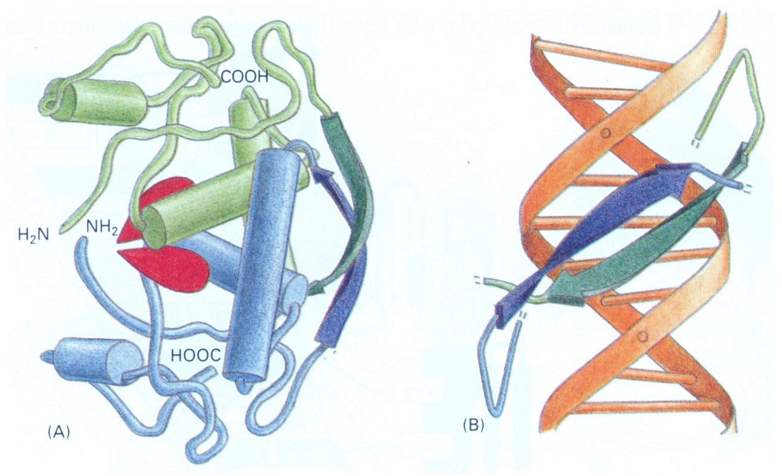
Riboswitch alternative



SAM

Grundy & Henkin, Mol. Microbiol 1998
Epshtein, et al., PNAS 2003
Winkler et al., Nat. Struct. Biol. 2003

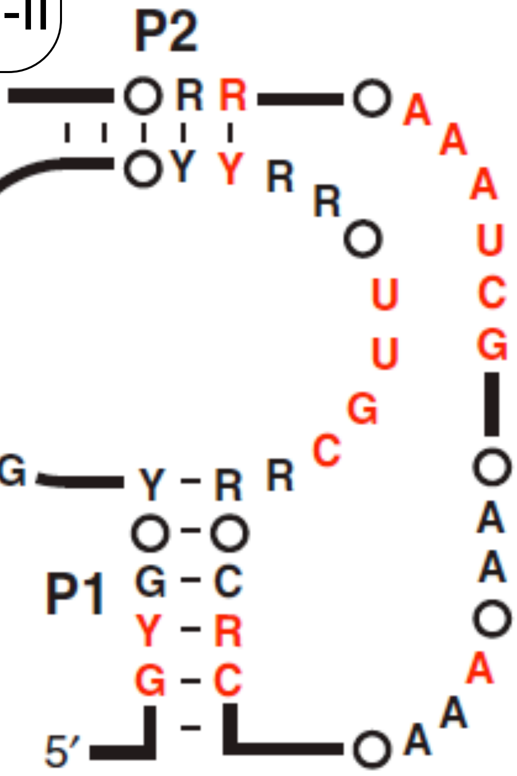
Alberts, et al, 3e.



← The protein way

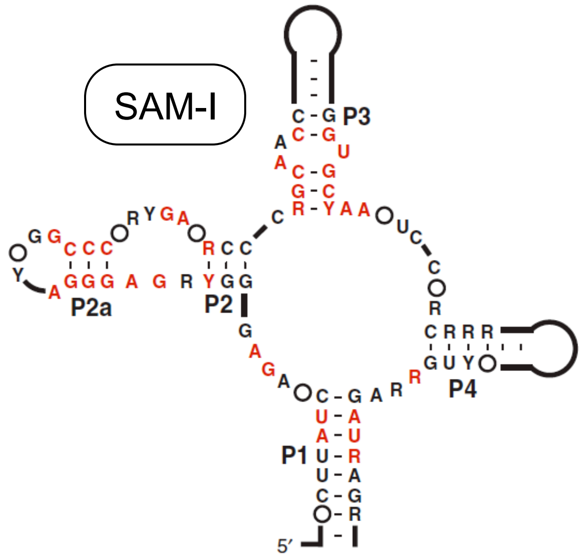
Riboswitch alternatives ↓

SAM-II



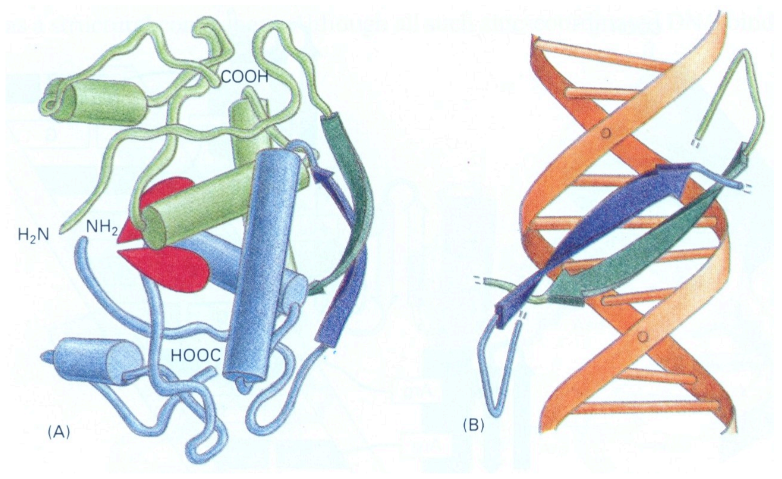
Corbino et al., Genome Biol. 2005

SAM-I



Grundy, Epshtein, Winkler et al., 1998, 2003

Alberts, et al, 3e.

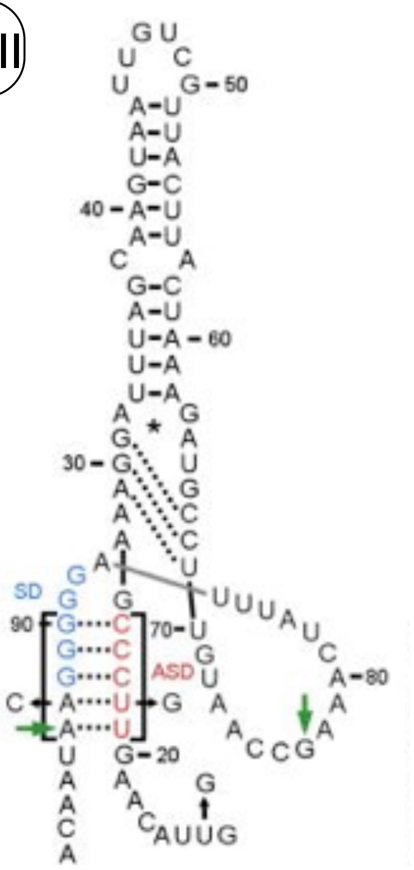


← The protein way

Riboswitch alternatives

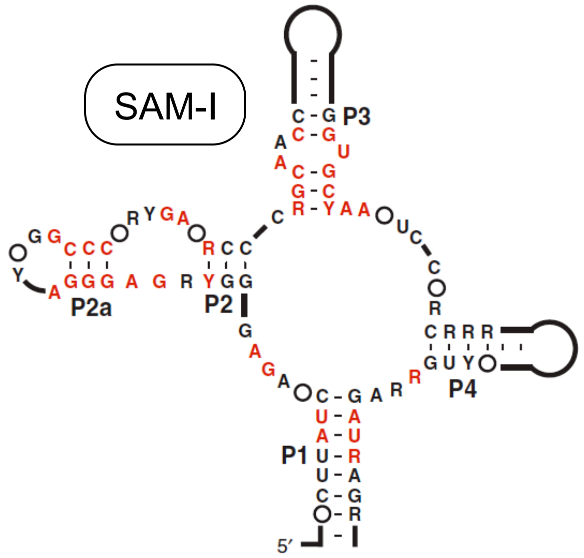
↓

SAM-III



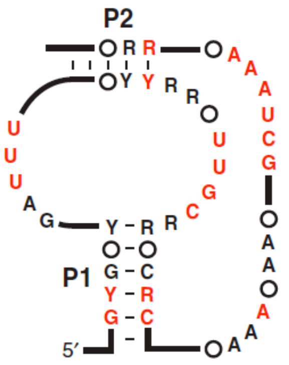
Fuchs et al., NSMB 2006

SAM-I



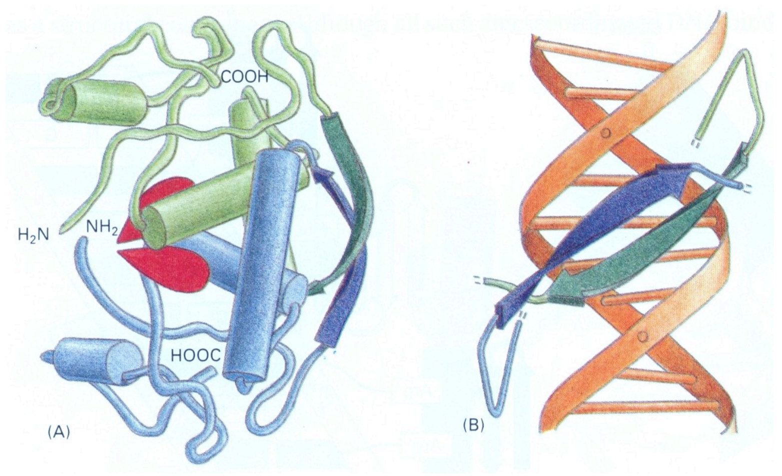
Grundy, Epshtein, Winkler et al., 1998, 2003

SAM-II



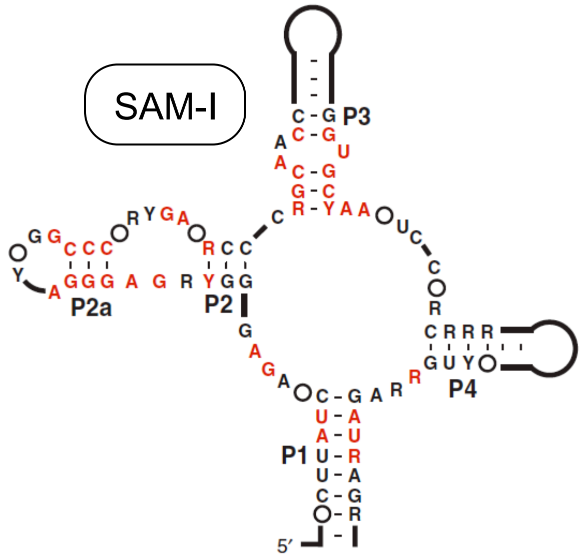
Corbino et al., Genome Biol. 2005

Alberts, et al, 3e.

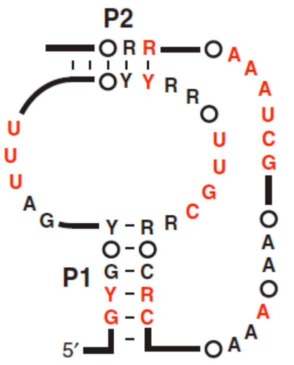


← The protein way

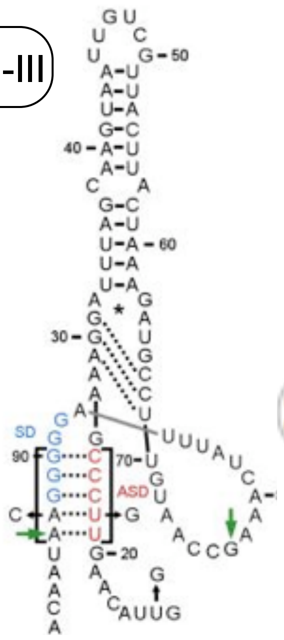
Riboswitch alternatives ↓



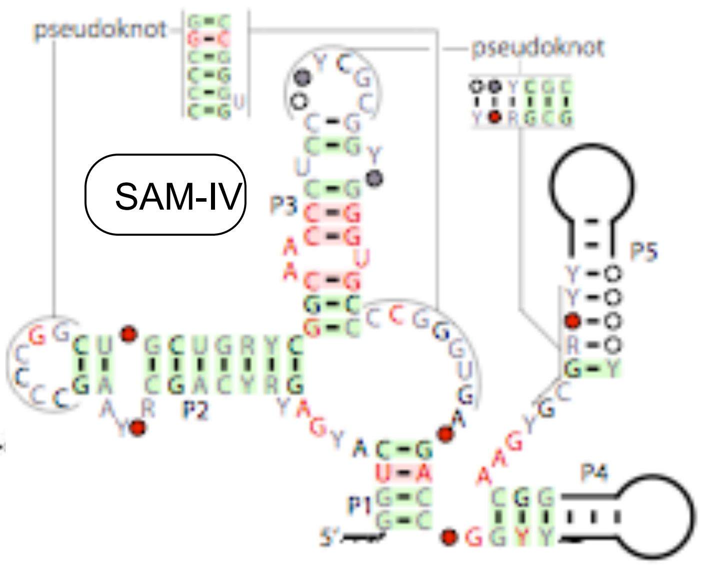
Grundy, Epshtein, Winkler et al., 1998, 2003



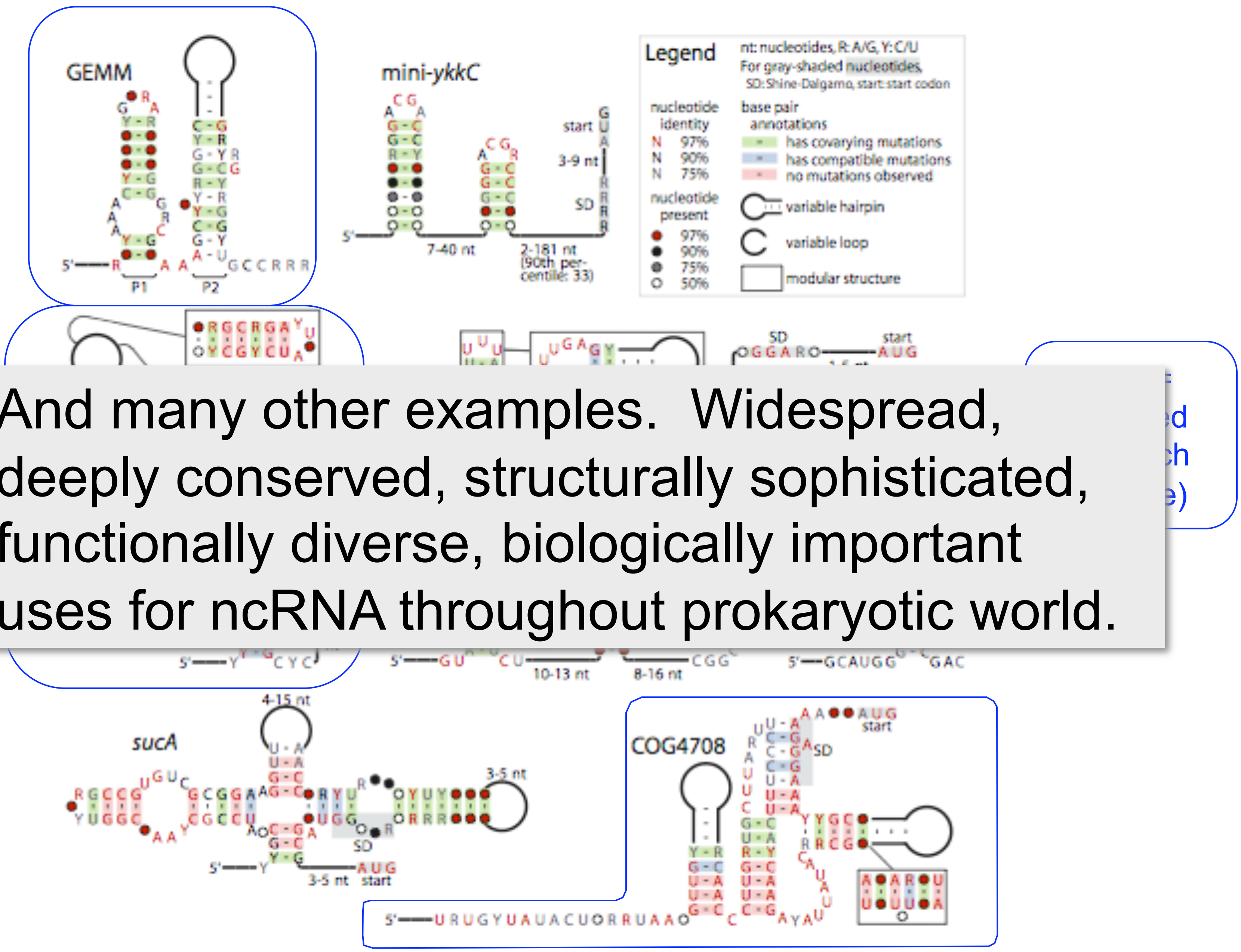
Corbino et al., Genome Biol. 2005



Fuchs et al., NSMB 2006



Weinberg et al., RNA 2008



And many other examples. Widespread, deeply conserved, structurally sophisticated, functionally diverse, biologically important uses for ncRNA throughout prokaryotic world.

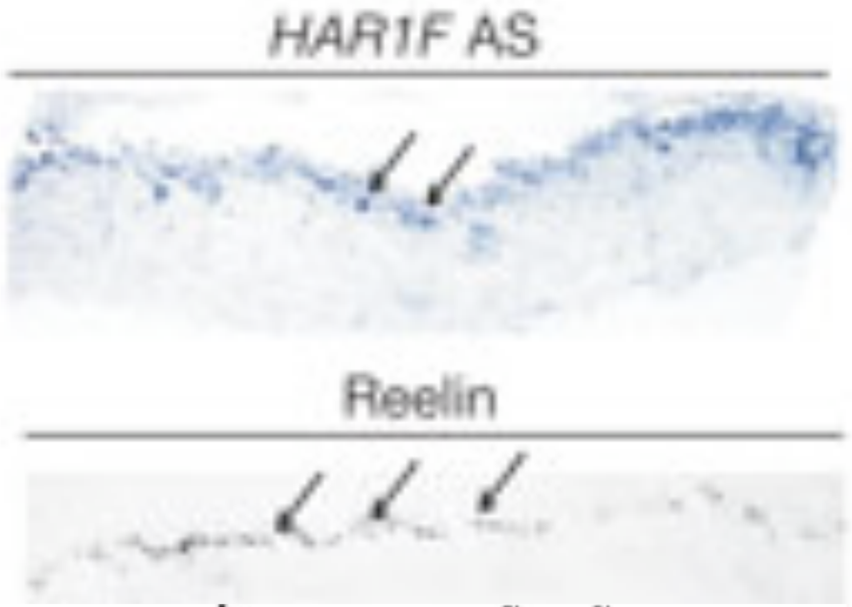
Weinberg, et al. Nucl. Acids Res., July 2007 35: 4809-4819.

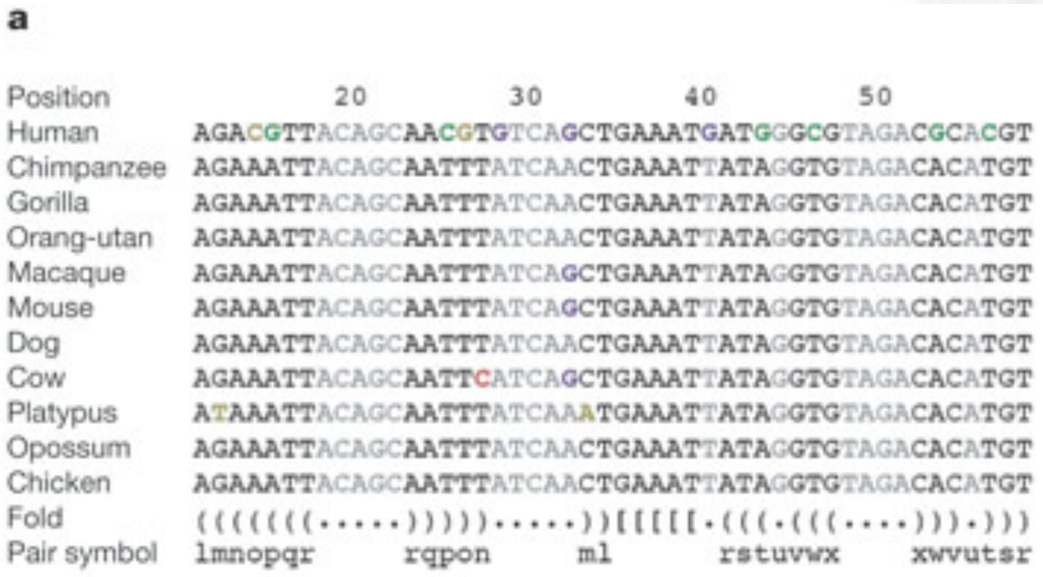
Vertebrates

- Bigger, more complex genomes
- <2% coding
- But >5% conserved in sequence?
- And 50-90% transcribed?
- And *structural* conservation, if any, invisible (without proper alignments, etc.)

- What’s going on?

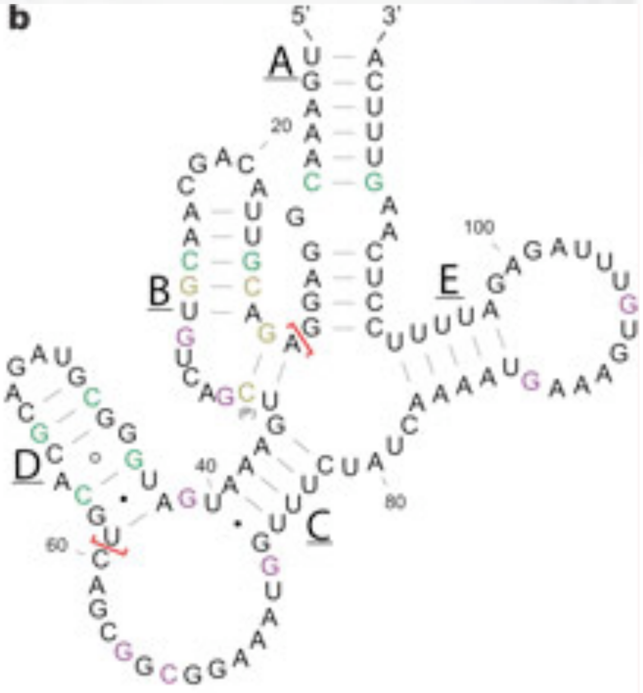
Fastest Human Gene?



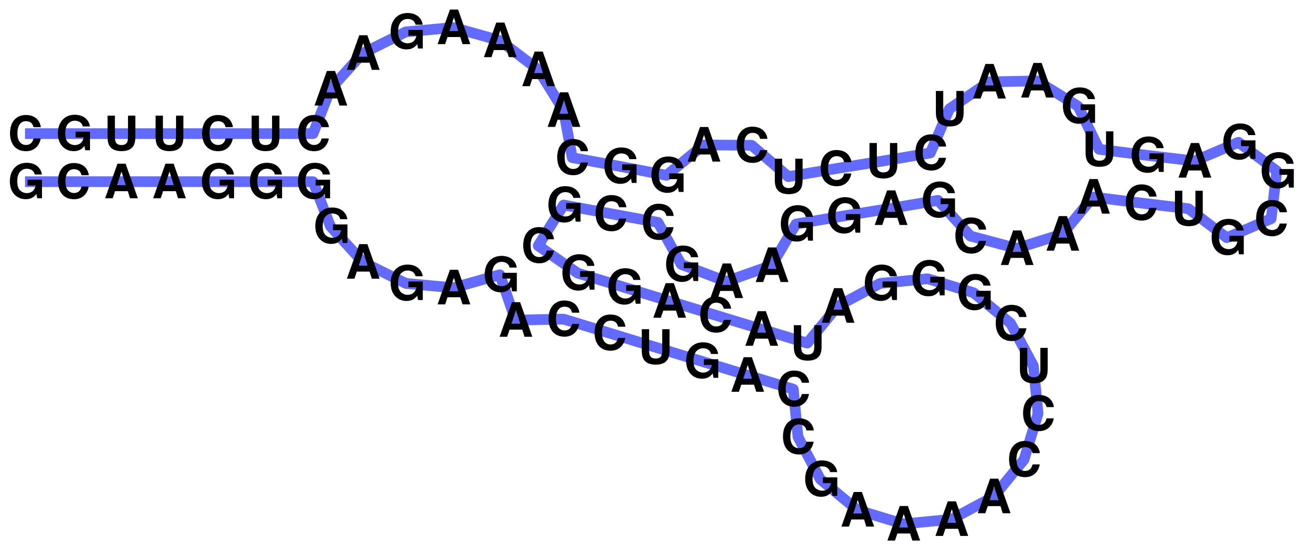


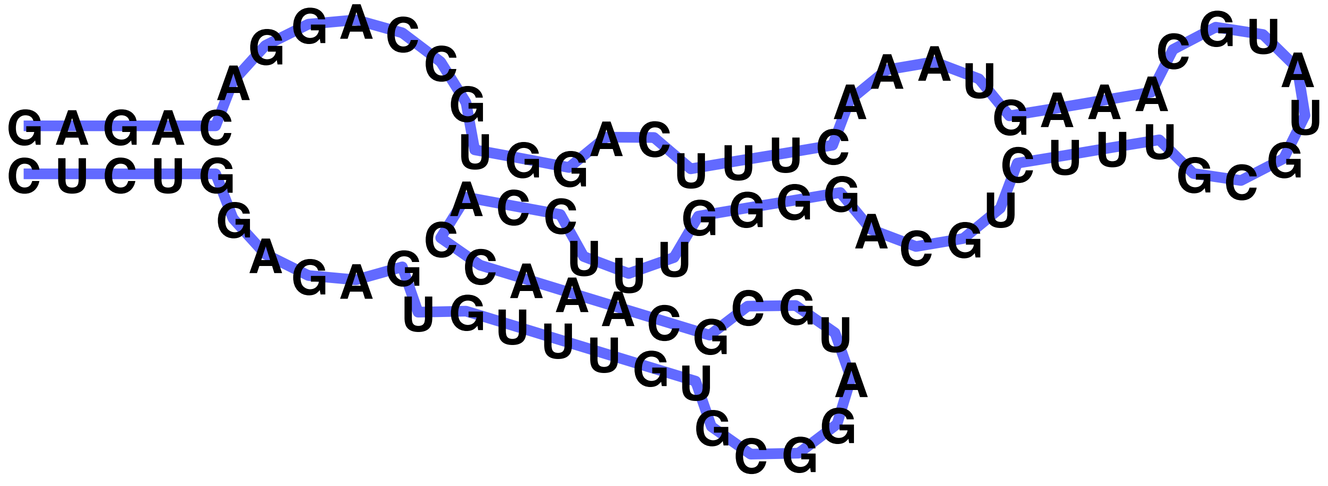
a

```
Position              20        30        40        50
Human        AGACGTTACAGCAACGTGTCAGCTGAAATGATGGGCGTAGACGCACGT
Chimpanzee   AGAAATTACAGCAATTTATCAACTGAAATTATAGGTGTAGACACATGT
Gorilla      AGAAATTACAGCAATTTATCAACTGAAATTATAGGTGTAGACACATGT
Orang-utan   AGAAATTACAGCAATTTATCAACTGAAATTATAGGTGTAGACACATGT
Macaque      AGAAATTACAGCAATTTATCAGCTGAAATTATAGGTGTAGACACATGT
Mouse        AGAAATTACAGCAATTTATCAGCTGAAATTATAGGTGTAGACACATGT
Dog          AGAAATTACAGCAATTTATCAACTGAAATTATAGGTGTAGACACATGT
Cow          AGAAATTACAGCAATTCATCAGCTGAAATTATAGGTGTAGACACATGT
Platypus     ATAAATTACAGCAATTTATCAAATGAAATTATAGGTGTAGACACATGT
Opossum      AGAAATTACAGCAATTTATCAACTGAAATTATAGGTGTAGACACATGT
Chicken      AGAAATTACAGCAATTTATCAACTGAAATTATAGGTGTAGACACATGT
Fold         (((((((.....))))).....))[[[[[.(((.(((....))).)))
Pair symbol  lmnopqr     rqpon     ml     rstuvwx     xwvutsr
```


b

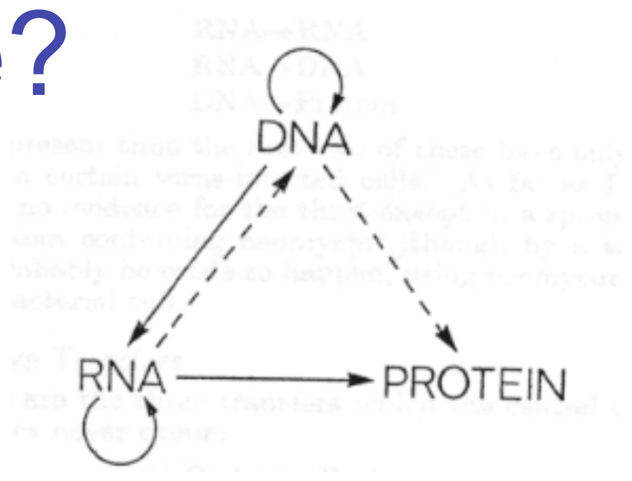
Q: What's so hard?





A: Structure often more important than sequence

Origin of Life?



Life needs
- information carrier: DNA
- molecular machines, like enzymes: Protein
- making proteins needs DNA + RNA + proteins
- making (duplicating) DNA needs proteins

Horrible circularities! How could it have arisen in an abiotic environment?

Origin of Life?

RNA can carry information, too

- RNA double helix; RNA-directed RNA polymerase

RNA can form complex structures

RNA enzymes exist (ribozymes)

RNA can control, do logic (riboswitches)

The “RNA world” hypothesis:
1st life was RNA-based

6.5 RNA Secondary Structure

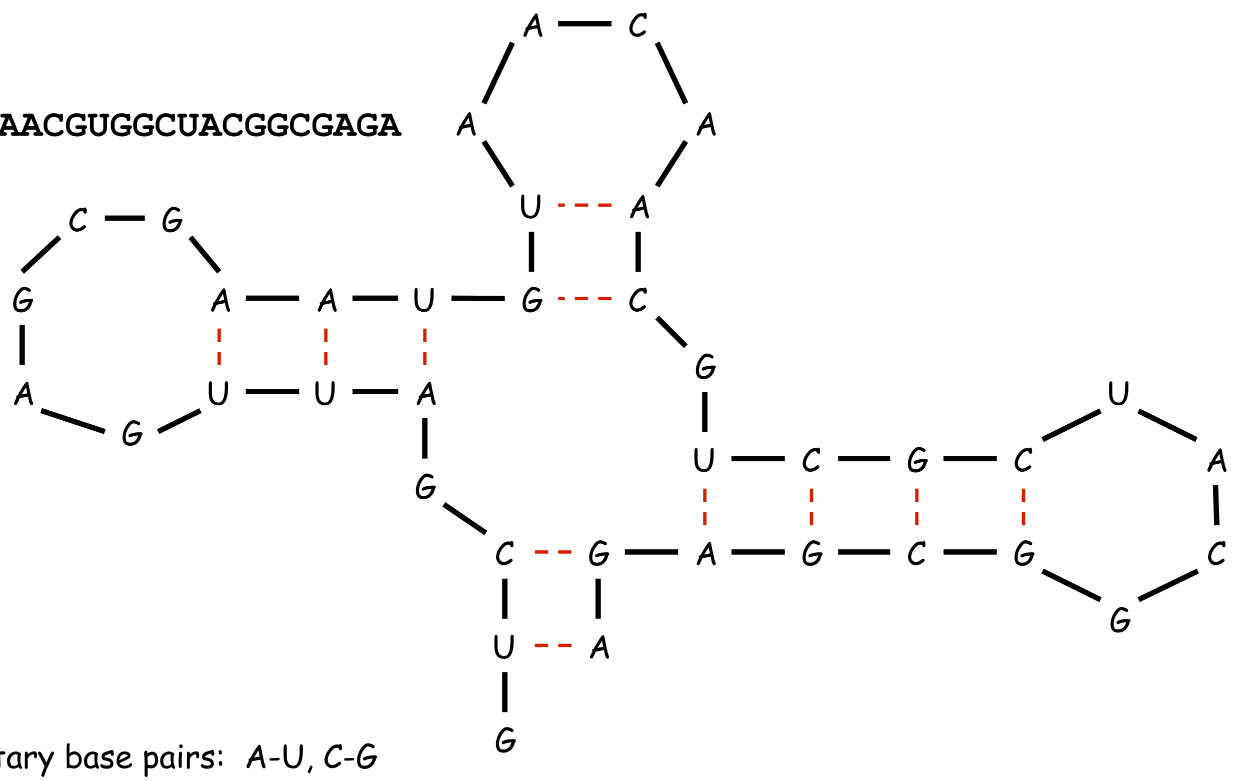
Nussinov's Algorithm – core technology for RNA structure prediction

RNA Secondary Structure

RNA. String B = $b_1 b_2 \ldots b_n$ over alphabet { A, C, G, U }.

Secondary structure. RNA is usually single-stranded, and tends to loop back and form base pairs with itself. This structure is essential for understanding behavior of molecule.

Ex: **GUCGAUUGAGCGAAUGUAACAACGUGGCUACGGCGAGA**



complementary base pairs: A-U, C-G

RNA Secondary Structure (somewhat oversimplified)

Secondary structure. A set of pairs $S = \{ (b_i, b_j) \}$ that satisfy:

- [Watson-Crick.]
  - S is a *matching* , i.e. each base pairs with at most one other, and
  - each pair in S is a Watson-Crick pair: A-U, U-A, C-G, or G-C.
- [No sharp turns.] The ends of each pair are separated by at least 4 intervening bases. If $(b_i, b_j) \in S$, then $i < j - 4$.
- [Non-crossing.] If (b_i, b_j) and (b_k, b_l) are two pairs in S, then we cannot have $i < k < j < l$. (Violation of this is called a *pseudoknot.*)

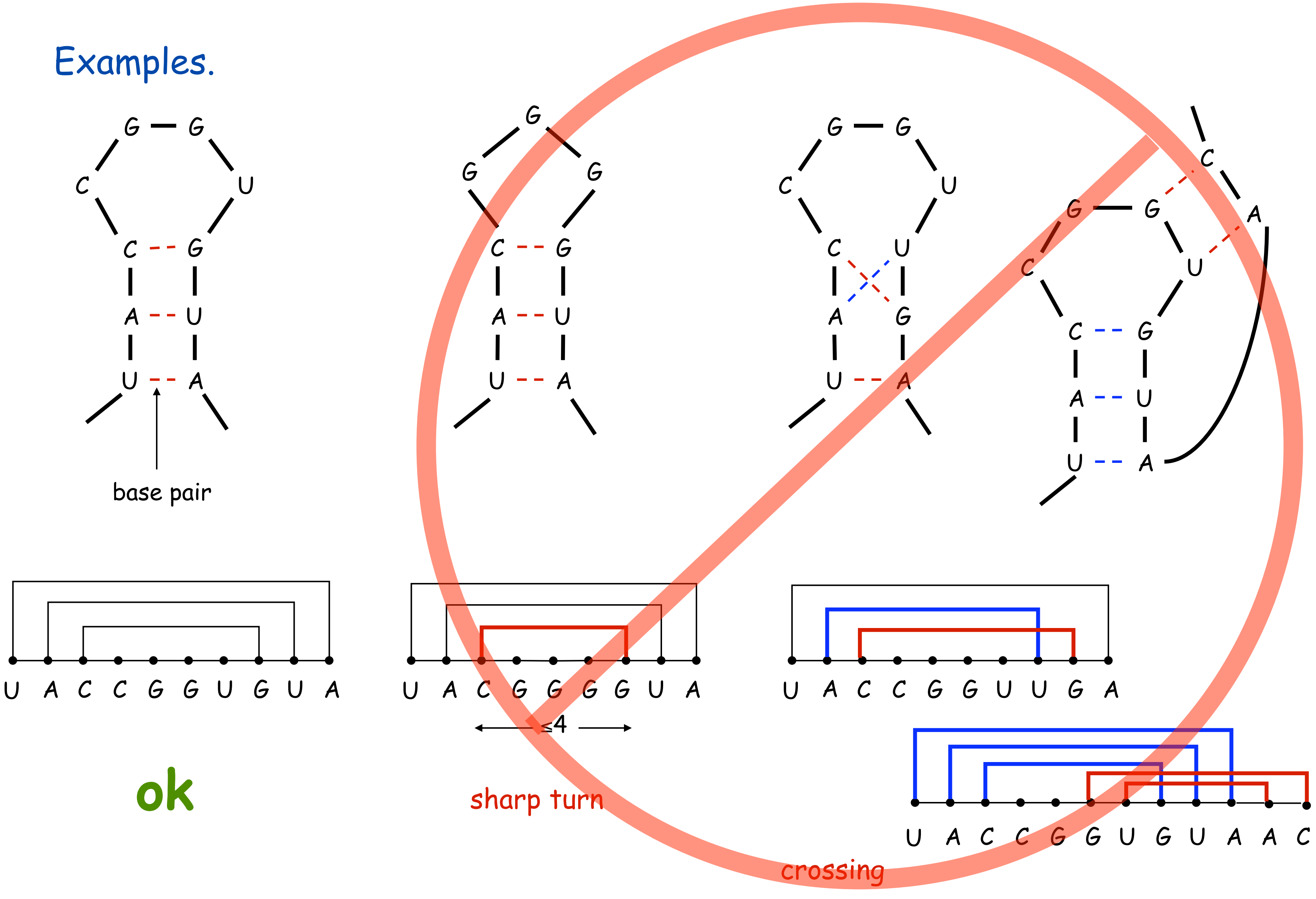
Free energy. Usual hypothesis is that an RNA molecule will form the secondary structure with the optimum total free energy.

approximate by number of base pairs

Goal. Given an RNA molecule $B = b_1 b_2 \ldots b_n$, find a secondary structure S that maximizes the number of base pairs.

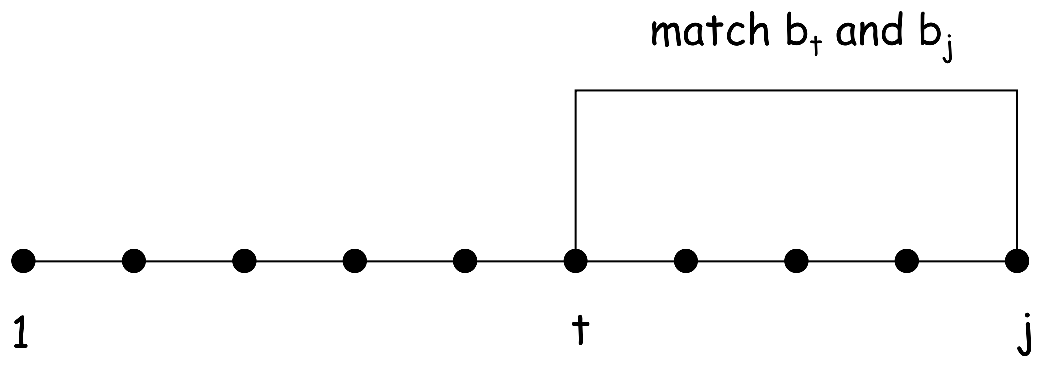
RNA Secondary Structure: Examples

Examples.



RNA Secondary Structure: Subproblems

First attempt. OPT[j] = maximum number of base pairs in a secondary structure of the substring $b_1b_2\ldots b_j$.



Difficulty. Results in two sub-problems.

- Finding secondary structure in: $b_1b_2\ldots b_{t-1}$. $\leftarrow$ OPT(t-1)
- Finding secondary structure in: $b_{t+1}b_{t+2}\ldots b_{j-1}$. $\leftarrow$ not OPT of anything; need more sub-problems

Dynamic Programming Over Intervals: (R. Nussinov's algorithm)

Notation. OPT[i, j] = maximum number of base pairs in a secondary structure of the substring $b_i b_{i+1} \ldots b_j$.

- Case 1. If $i \geq j - 4$.
  - OPT[i, j] = 0 by no-sharp turns condition.
- Case 2. Base b_j is not involved in a pair.
  - OPT[i, j] = OPT[i, j-1]
- Case 3. Base b_j pairs with b_t for some $i \leq t < j - 4$.
  - non-crossing constraint decouples resulting sub-problems
  - $\text{OPT}[i, j] = 1 + \underset{\uparrow}{\max_t} \{ \text{OPT}[i, t-1] + \text{OPT}[t+1, j-1] \}$

take max over t such that $i \leq t < j-4$ and b_t and b_j are Watson-Crick complements

Key point: Either last base is unpaired (case 1,2) or paired (case 3)

Remark. Same core idea in CKY algorithm to parse context-free grammars.

Bottom Up Dynamic Programming Over Intervals

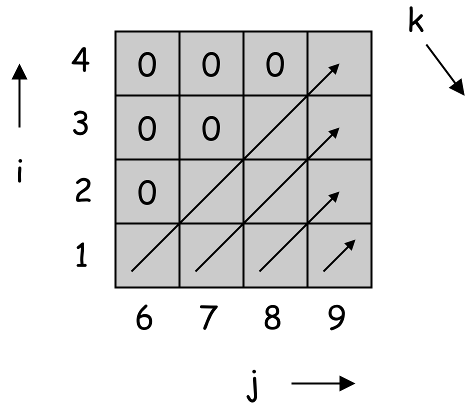
Q. What order to solve the sub-problems?

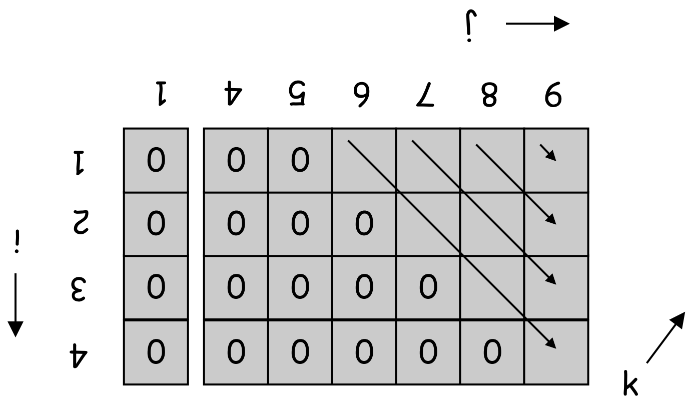
A. Do shortest intervals first.

```
RNA(b_1,…,b_n) {
   for k = 5, 6, …, n-1
      for i = 1, 2, …, n-k
         j = i + k
         Compute OPT[i, j]

   return OPT[1, n]
}
```

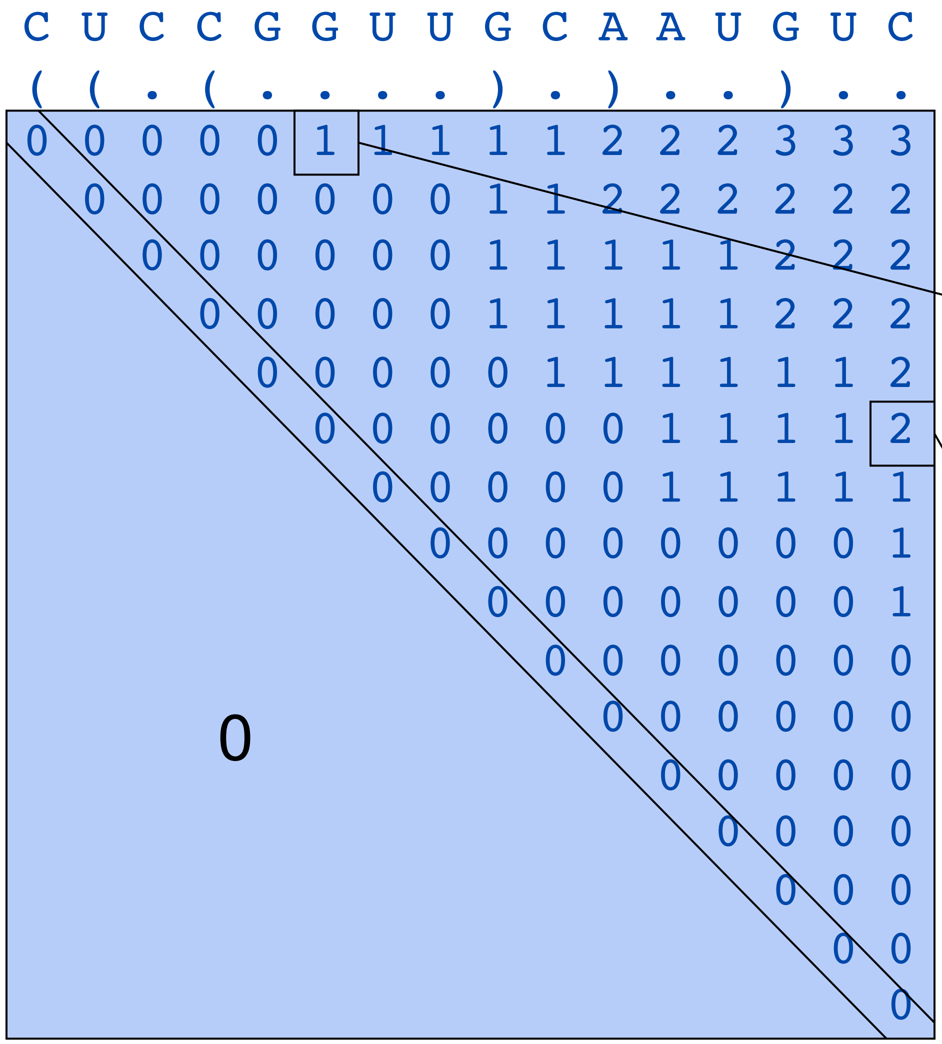
using recurrence





Running time. $O(n^3)$.

n = 16



C	U	C	C	G	G	U	U	G	C	A	A	U	G	U	C
(	(	.	(	.	.	.	.	)	.	)	.	.	)	.	.
0	0	0	0	0	1	1	1	1	1	2	2	2	3	3	3
	0	0	0	0	0	0	0	1	1	2	2	2	2	2	2
		0	0	0	0	0	0	1	1	1	1	1	2	2	2
			0	0	0	0	0	1	1	1	1	1	2	2	2
				0	0	0	0	0	1	1	1	1	1	1	2
					0	0	0	0	0	0	1	1	1	1	2
						0	0	0	0	0	1	1	1	1	1
							0	0	0	0	0	0	0	0	1
								0	0	0	0	0	0	0	1
									0	0	0	0	0	0	0
										0	0	0	0	0	0
											0	0	0	0	0
												0	0	0	0
													0	0	0
														0	0
															0

E.g.:
OPT[1,6] = 1:

CUCCGG

(....)

E.g.:
OPT[6,16] = 2:

GUUGCAAUGUC

((....)...)

Computing one cell: OPT[2,18] = ?

G	G	G	A	A	A	A	C	C	C	A	A	A	G	G	G	G	$\boxed{U}$	U	U
(	(	(	.	.	.	.	)	)	)	(	(	(	.	.	.	.	)	)	)
0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	0	1	2	2	2	2	2	2	3	3	3	$\boxed{4}$	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

n= 20

Case 1:
2 ≥ 18-4? no.
Case 2:
B_{18} unpaired?
Always a possibility;
then OPT[2,18] ≥ 3

GGAAAACCCAAAGGGGU
((. . . .)) (. . . .) . . .

$$\mathrm{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\begin{Bmatrix} \mathrm{OPT}[i,j-1] \\ 1+\max_t(\mathrm{OPT}[i,t-1]+\mathrm{OPT}[t+1,j-1]) \end{Bmatrix} & \text{otherwise} \end{cases}$$

Computing one cell: OPT[2,18] = ?

n= 20

G	G	G	A	A	A	A	C	C	C	A	A	A	G	G	G	G	U	U	U
(	(	(	.	.	.	.	)	)	)	(	(	(	.	.	.	.	)	)	)
0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	0	1	2	2	2	2	2	2	3	3	3	4	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Case 3, 2 ≤ t <18-4:
t = 2: no pair

$$\mathrm{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\begin{Bmatrix} \mathrm{OPT}[i,j-1] \\ 1+\max_t(\mathrm{OPT}[i,t-1]+\mathrm{OPT}[t+1,j-1]) \end{Bmatrix} & \text{otherwise} \end{cases}$$

Computing one cell: OPT[2,18] = ?

n= 20

G	G	G	A	A	A	A	C	C	C	A	A	A	G	G	G	G	U	U	U
(	(	(	.	.	.	.	)	)	)	(	(	(	.	.	.	.	)	)	)
0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	0	1	2	2	2	2	2	2	3	3	3	4	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Case 3, 2 ≤ t <18-4:
t = 3: no pair

$$\mathrm{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\begin{Bmatrix} \mathrm{OPT}[i,j-1] \\ 1+\max_t(\mathrm{OPT}[i,t-1]+\mathrm{OPT}[t+1,j-1]) \end{Bmatrix} & \text{otherwise} \end{cases}$$

Computing one cell: OPT[2,18] = ?

n= 20

G	G	G	A	A	A	A	C	C	C	A	A	A	G	G	G	G	U	U	U
(	(	(	.	.	.	.	)	)	)	(	(	(	.	.	.	.	)	)	)
0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	0	1	2	2	2	2	2	2	3	3	3	4	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

$$\text{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\begin{cases} \text{OPT}[i,j-1] \\ 1+\max_t(\text{OPT}[i,t-1]+\text{OPT}[t+1,j-1]) \end{cases} & \text{otherwise} \end{cases}$$

Case 3, 2 ≤ t <18-4:
 t = 4: yes pair
 OPT[2,18]≥1+0+3

GG**A**AAACCCAAAGGGG**U**
..(...((((....))))

Computing one cell: OPT[2,18] = ?

n= 20

G	G	G	A	A	A	A	C	C	C	A	A	A	G	G	G	G	U	U	U
(	(	(	.	.	.	.	)	)	)	(	(	(	.	.	.	.	)	)	)
0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	0	1	2	2	2	2	2	2	3	3	3	4	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Case 3, 2 ≤ t <18-4:
t = 5: yes pair
OPT[2,18]≥1+0+3

GGA**A**AACCCAAAGGGG**U**
...(..((((....))))

$$\mathrm{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\left\{ \begin{array}{c} \mathrm{OPT}[i,j-1] \\ 1+\max_t(\mathrm{OPT}[i,t-1]+\mathrm{OPT}[t+1,j-1]) \end{array} \right\} & \text{otherwise} \end{cases}$$

Computing one cell: OPT[2,18] = ?

n= 20

G	G	G	A	A	A	A	C	C	C	A	A	A	G	G	G	G	U	U	U
(	(	(	.	.	.	.	)	)	)	(	(	(	.	.	.	.	)	)	)
0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	0	1	2	2	2	2	2	2	3	3	3	4	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Case 3, 2 ≤ t <18-4:

t = 6: yes pair

OPT[2,18]≥1+0+3

GGAAA**A**ACCCAAAGGGG**U**

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

$$\text{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\begin{cases} \text{OPT}[i,j-1] \\ 1+\max_t(\text{OPT}[i,t-1]+\text{OPT}[t+1,j-1]) \end{cases} & \text{otherwise} \end{cases}$$

Computing one cell: OPT[2,18] = ?

n= 20

G	G	G	A	A	A	**A**	C	C	C	A	A	A	G	G	G	G	**U**	U	U
(	(	(	.	.	.	.	)	)	)	(	(	(	.	.	.	.	)	)	)
0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	0	1	2	2	2	2	2	2	3	3	3	4	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Case 3, $2 \le t < 18-4$:
$t = 7$: yes pair
OPT[2,18]$\ge 1+0+3$

GGAAA**A**CCCAAAGGGG**U**
.....((((....))))

$$\mathrm{OPT}(i,j) = \begin{cases} 0 & \text{if } i \ge j-4 \\ \max\begin{Bmatrix} \mathrm{OPT}[i,j-1] \\ 1+\max_t(\mathrm{OPT}[i,t-1]+\mathrm{OPT}[t+1,j-1]) \end{Bmatrix} & \text{otherwise} \end{cases}$$

Computing one cell: OPT[2,18] = ?

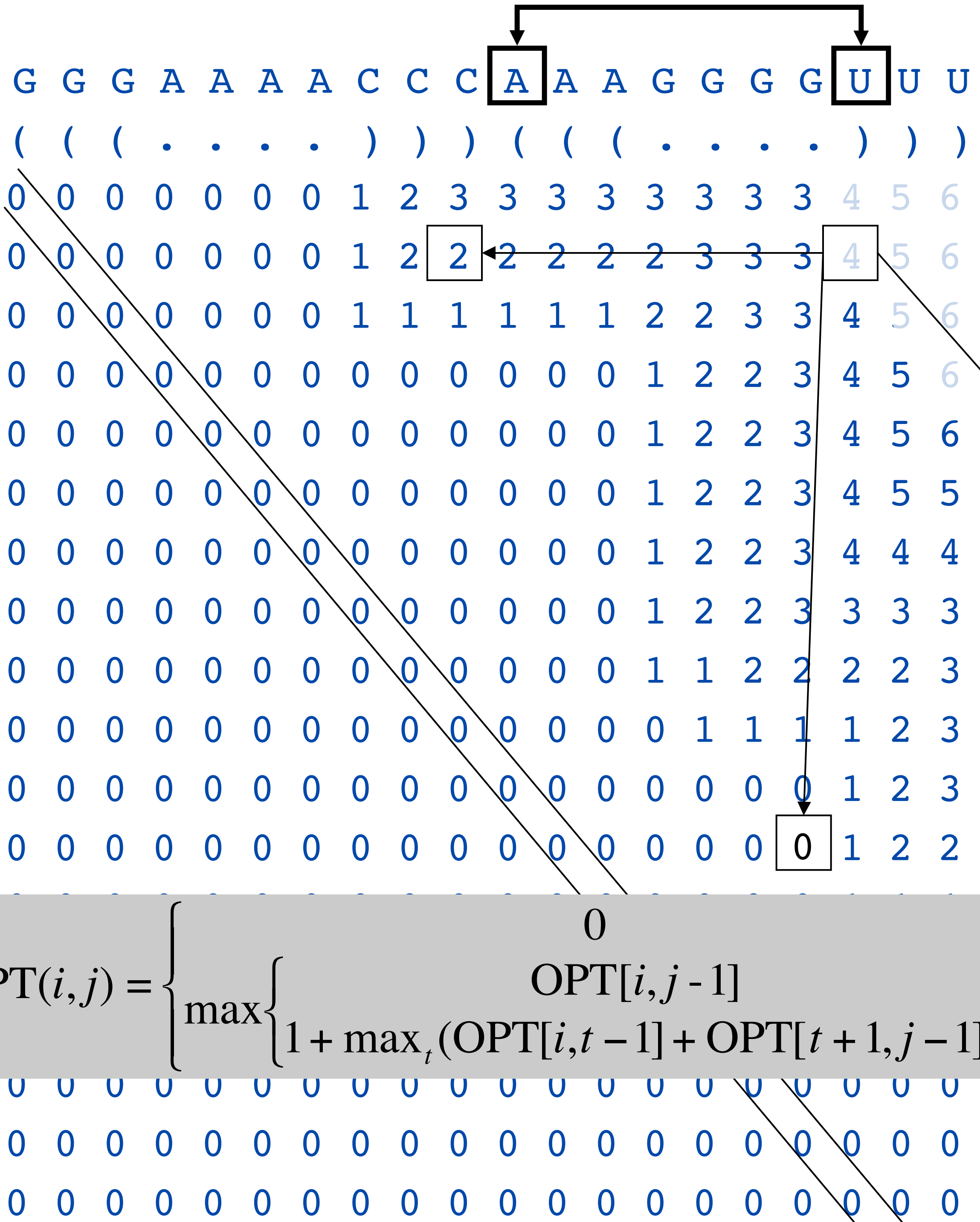
n= 20

G	G	G	A	A	A	A	$\boxed{C}$	C	C	A	A	A	G	G	G	G	$\boxed{U}$	U	U
(	(	(	.	.	.	.	)	)	)	(	(	(	.	.	.	.	)	)	)
0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	$\boxed{0}$	1	2	2	2	2	2	2	3	3	3	$\boxed{4}$	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	$\boxed{2}$	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Case 3, 2 ≤ t <18-4:
t = 8: no pair

$$\mathrm{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\left\{\begin{array}{c} \mathrm{OPT}[i,j-1] \\ 1+\max_t(\mathrm{OPT}[i,t-1]+\mathrm{OPT}[t+1,j-1]) \end{array}\right\} & \text{otherwise} \end{cases}$$

Computing one cell: OPT[2,18] = ?



G G G A A A A C C C A A A G G G G U U U

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

n= 20

0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	0	1	2	2	2	2	2	2	3	3	3	4	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

$$\text{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\begin{Bmatrix} \text{OPT}[i,j-1] \\ 1+\max_t(\text{OPT}[i,t-1]+\text{OPT}[t+1,j-1]) \end{Bmatrix} & \text{otherwise} \end{cases}$$

Case 3, 2 ≤ t <18-4:

t = 11: yes pair

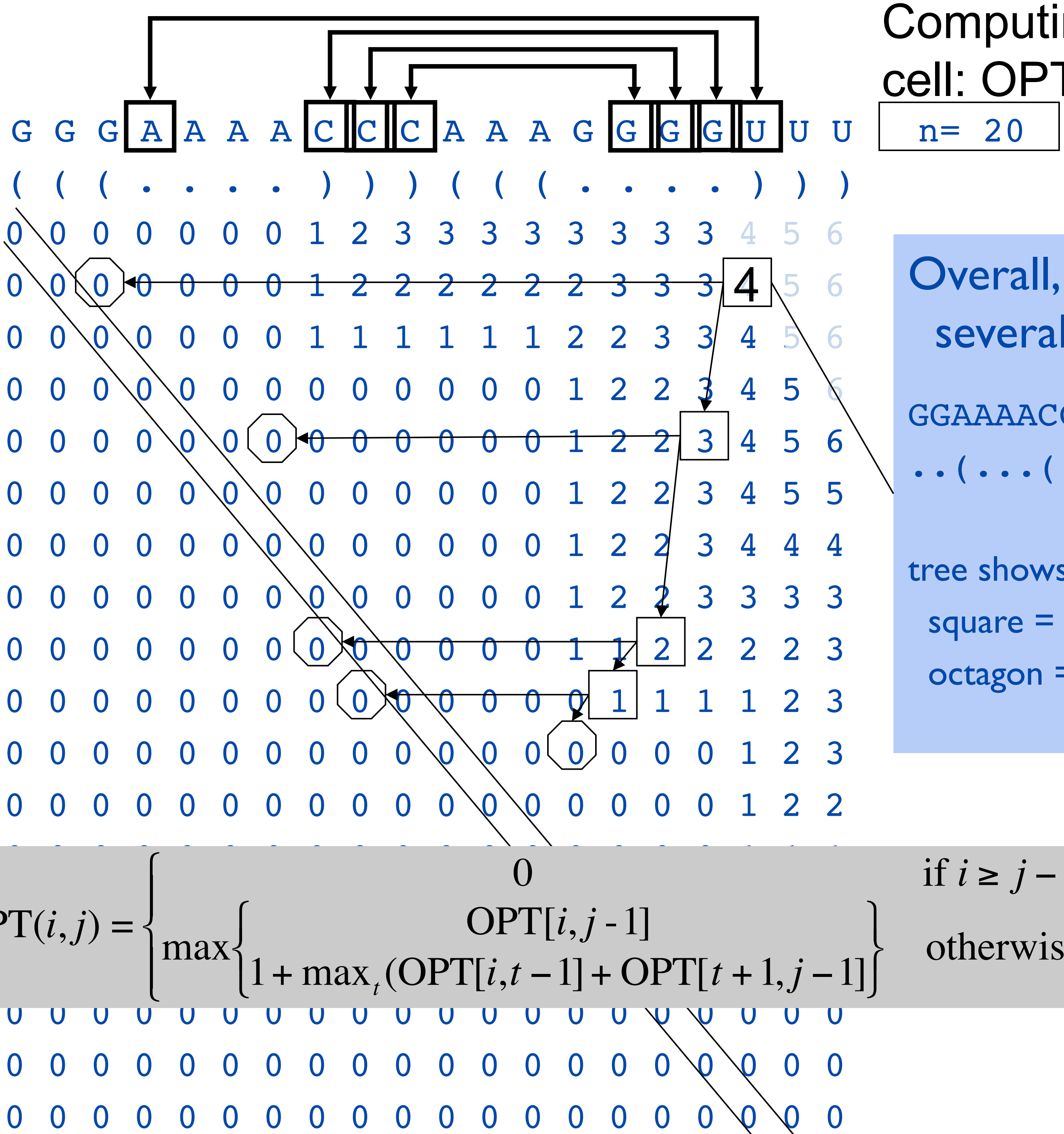
OPT[2,18]≥1+2+0

GGAAAACCC**A**AAGGGG**U**

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

(not shown: t=9,10, 12,13)

Computing one cell: OPT[2,18] = 4



0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3	3	4	5	6
0	0	0	0	0	0	0	1	2	2	2	2	2	2	3	3	3	4	5	6
0	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	6
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	5	5
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	3	3	3	3
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

$$\mathrm{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\begin{cases} \mathrm{OPT}[i,j-1] \\ 1+\max_t(\mathrm{OPT}[i,t-1]+\mathrm{OPT}[t+1,j-1]) \end{cases} & \text{otherwise} \end{cases}$$

Overall, Max = 4 several ways, e.g.:

GGAAAACCCAAAGGGGU

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

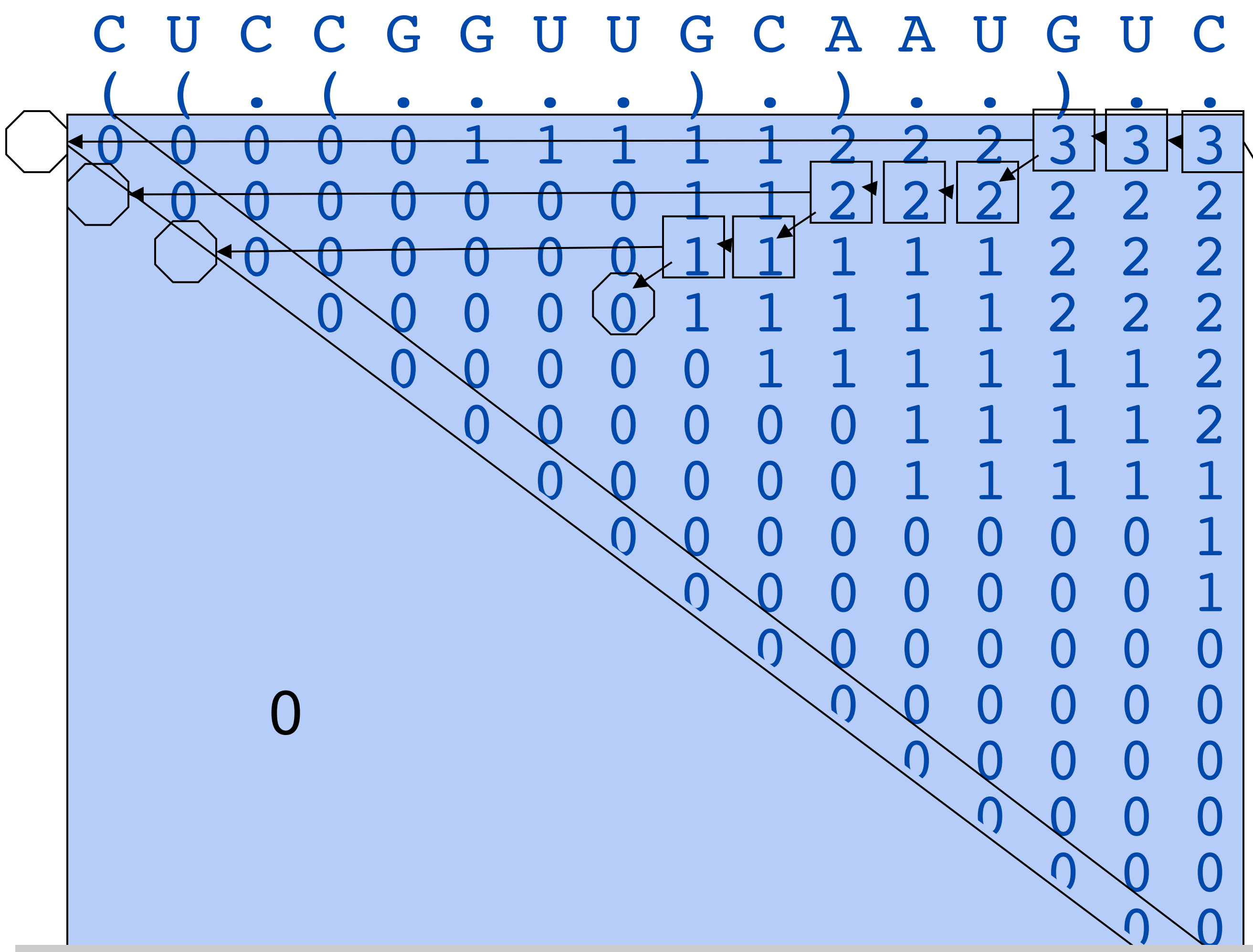
tree shows trace back:

square = case 3

octagon = case 1

Another Trace Back Example

n = 16



C	U	C	C	G	G	U	U	G	C	A	A	U	G	U	C
(	(	.	(	.	.	.	.	)	.	)	.	.	)	.	.
0	0	0	0	0	1	1	1	1	1	2	2	2	3	3	3
	0	0	0	0	0	0	0	1	1	2	2	2	2	2	2
		0	0	0	0	0	0	1	1	1	1	1	2	2	2
			0	0	0	0	0	1	1	1	1	1	2	2	2
				0	0	0	0	0	1	1	1	1	1	1	2
					0	0	0	0	0	0	1	1	1	1	2
						0	0	0	0	0	1	1	1	1	1
							0	0	0	0	0	0	0	0	1
								0	0	0	0	0	0	0	1
									0	0	0	0	0	0	0
										0	0	0	0	0	0
											0	0	0	0	0
												0	0	0	0
													0	0	0
														0	0
															0

E.g.:
OPT[1,16] = 3:

CUCCGGUUGCAAUGUC

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

$$\mathrm{OPT}(i,j) = \begin{cases} 0 & \text{if } i \geq j-4 \\ \max\begin{Bmatrix} \mathrm{OPT}[i,j-1] \\ 1+\max_t(\mathrm{OPT}[i,t-1]+\mathrm{OPT}[t+1,j-1]) \end{Bmatrix} & \text{otherwise} \end{cases}$$