CSI5126. Algorithms in bioinformatics

RNA Secondary Structure **Inference**

Marcel Turcotte

School of Electrical Engineering and Computer Science (EECS)
University of Ottawa

Version November 15, 2018

Summary

RNA (secondary) structure will be the focus of this module. We learn that RNA evolves so as to preserve bair pairs patterns more than sequence. We discuss the impact on traditional bioinformatics approaches. Today's lecture focuses on the **inference problem**, whereas the next lecture will be about the **search problem**.

General objective

- **Implement** the Nussinov algorithm for finding a secondary structure maximising the number of base pairs that can be formed in an input RNA sequence.

Reading

- Wing-Kin Sung (2010) Algorithms in Bioinformatics: A Practical Introduction. Chapman & Hall/CRC. QH 324.2 .S86 2010, Chapter 11.

Take home message

- With RNAs, **base pair patterns** are more preserved than **sequence**

Take home message

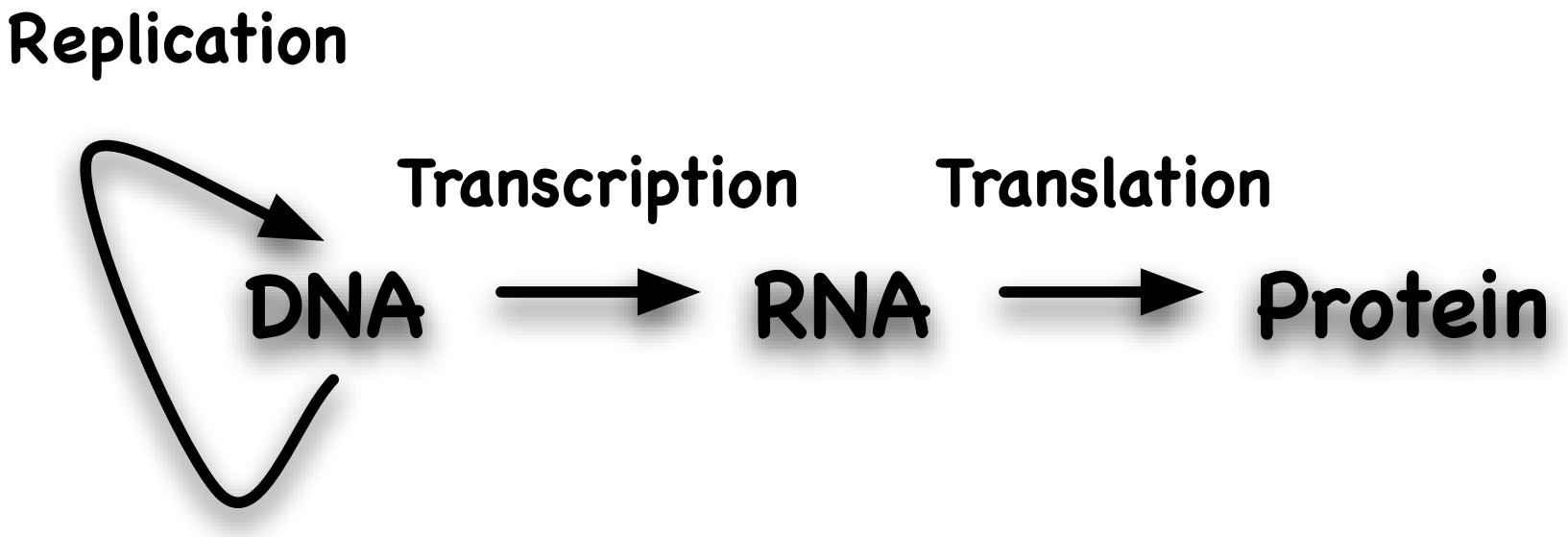
- With RNAs, **base pair patterns** are more preserved than **sequence**
- Consequently, **traditional bioinformatics** tools are generally not well adapted to RNA research

The **other** take home **message**

- "It is impossible to **understand** the **biology** of multicellular organisms without appreciation of the roles that **small RNAs** play."

Neilson, J. R., & Sharp, P. A. (2008). Small RNA regulators of gene expression. *Cell*, **134**(6), 899–902. `http://doi.org/10.1016/j.cell.2008.09.006`

Central Dogma



Non-coding RNA (aka non-protein-coding RNA, RNA gene, functional RNA) is a **transcribed** RNA molecule that is **not translated** into a protein.

Time line

- **1953** — DNA double-helix

Time line

- **1953** — DNA double-helix
- **1976** — Transfer RNA crystal structure

Time line

- **1953** — DNA double-helix
- **1976** — Transfer RNA crystal structure
- **1989** — Cech and Altman get a Nobel price in chemistry for the discovery of catalytic RNAs

Time line

- **1953** — DNA double-helix
- **1976** — Transfer RNA crystal structure
- **1989** — Cech and Altman get a Nobel price in chemistry for the discovery of catalytic RNAs
- **2000** — X-ray structure of the large ribosomal subunits

Time line

- **1953** — DNA double-helix
- **1976** — Transfer RNA crystal structure
- **1989** — Cech and Altman get a Nobel price in chemistry for the discovery of catalytic RNAs
- **2000** — X-ray structure of the large ribosomal subunits
- **2006** — Fire and Mello get a Nobel prize in physiology or medicine for their discovery of RNA interference

Concepts

- RNA has **structural complexity** rivaling proteins

Concepts

- RNA has **structural complexity** rivaling proteins
- RNA molecules are the major players of the genetic code: **message**, **splicing**, **transfer**, **regulation**

Concepts

- RNA has **structural complexity** rivaling proteins
- RNA molecules are the major players of the genetic code: **message**, **splicing**, **transfer**, **regulation**
- RNA have been discovered in many other systems: DNA packaging, telomeres, **editing**, etc.

Concepts

- RNA has **structural complexity** rivaling proteins
- RNA molecules are the major players of the genetic code: **message**, **splicing**, **transfer**, **regulation**
- RNA have been discovered in many other systems: DNA packaging, telomeres, **editing**, etc.
- **RNA therapeutics:** antisense, ribozymes, RNA beacons

RNA **Catalyses** Reaction

- Ribozymes (**RNA enzymes**) discovery, early 1980s
- The Nobel Prize in Chemistry 1989

Thomas R. **Cech** (Colorado)

Sidney **Altman** [a][b] (Yale)

[a] Born in Montréal
[b] Guest member uOttawa OISB

Examples of catalytic RNAs (Ribozymes)

- **Ribonuclease P (RNase P)** is a ribo-protein complex responsible for removing (cleaving off) an extra sequence of pre-tRNA in the process of tRNA maturation. The RNA component is the catalyst.
- **Group I and II introns** are self splicing (auto-catalytic).
- Many **artificial ribozymes** have been produced by an experiment called SELEX (Systematic Evolution of Ligands by Exponential Enrichment).

RNA World

- 1986, **RNA World hypothesis**
- RNA has the ability to store information, as DNA does
- RNA has the ability to catalyze reactions, as proteins do
- RNA is an ideal candidate for an earlier simple form of life

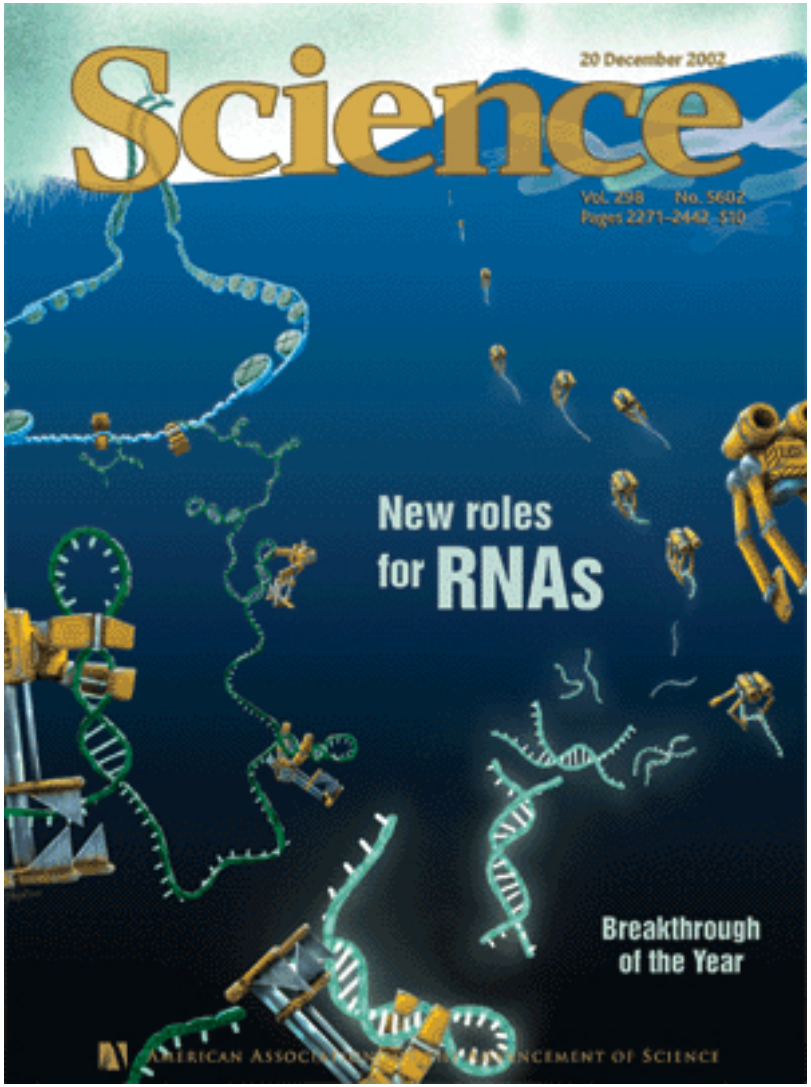
Walter **Gilbert**
(Nobel Prize in Chemistry 1980)

RNA World

> *The phrase "**The RNA World**" was coined by Walter Gilbert in 1986 in a commentary on the then recent observations of the **catalytic** properties of various RNAs. The RNA World referred to an **hypothetical stage in the origin of life on Earth**. During this stage, proteins were not yet engaged in biochemical reactions and **RNA carried out both the information storage task of genetic information and the full range of catalytic roles necessary in a very primitive self-replicating system**.*

nobelprize.org/nobel_prizes/chemistry/articles/altman
(Visited November 7, 2006)

Small RNAs as 2002 Science Breakthrough



"Researchers are discovering that small RNA molecules play a surprising variety of key roles in cells. They can **inhibit translation of messenger RNA into protein**, **cause degradation of other messenger RNAs**, and **even initiate complete silencing of gene expression** from the genome."

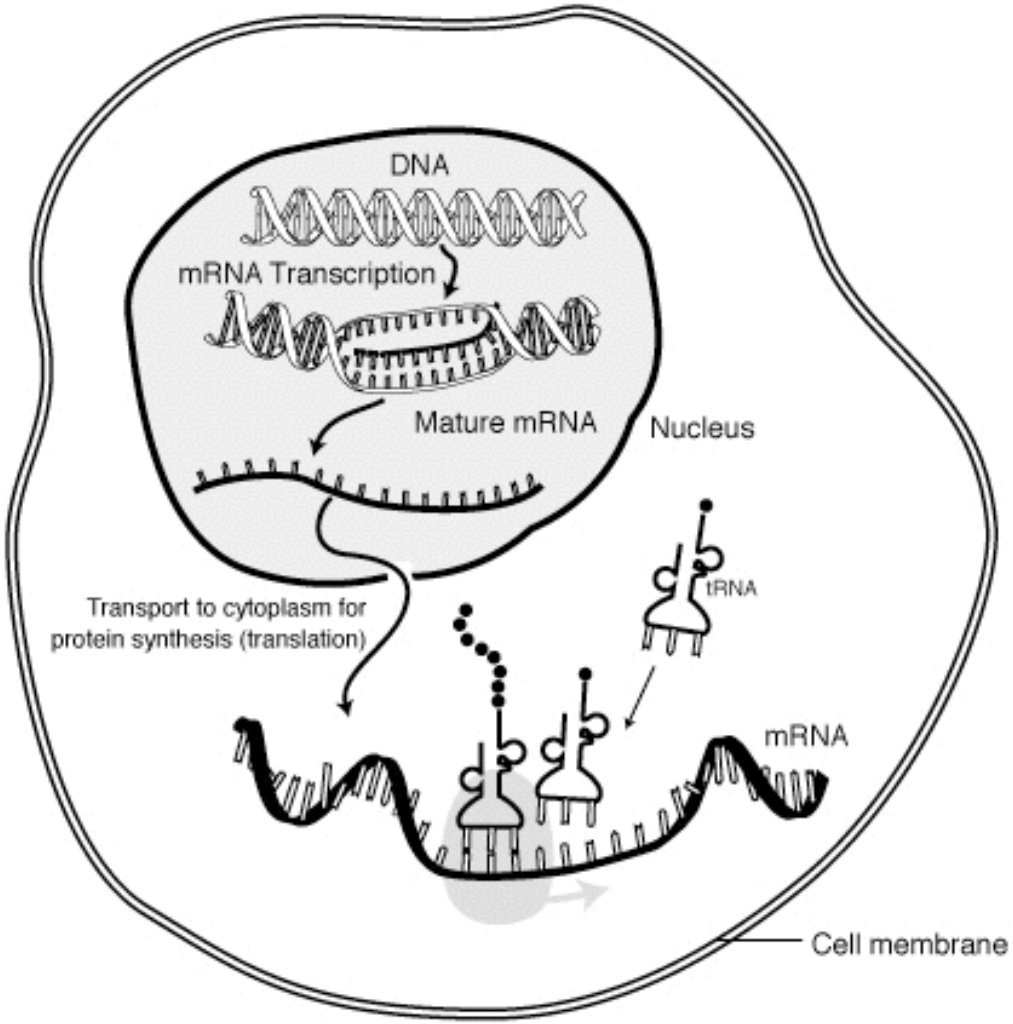
RNA **Controls** Gene Expression

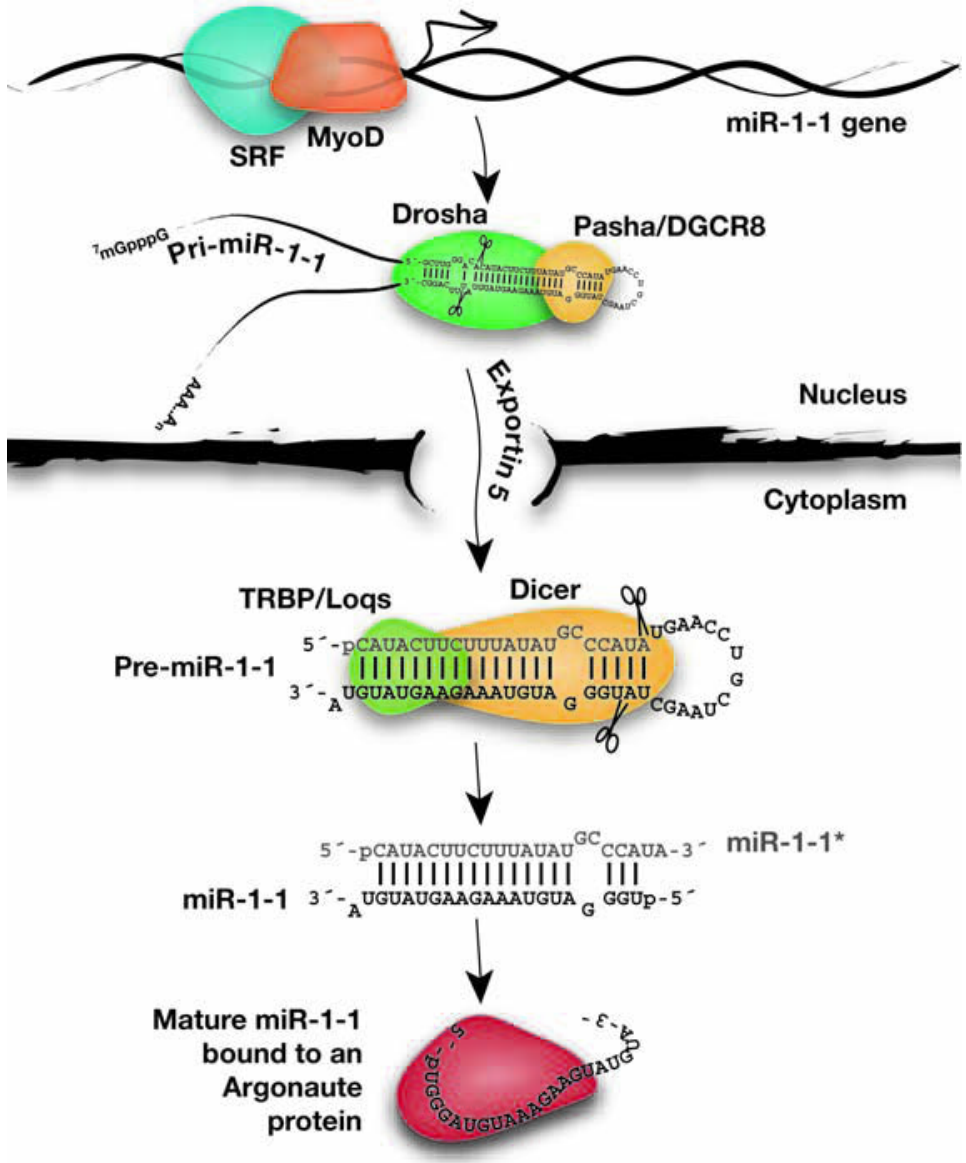
- The Nobel Prize in Physiology or Medicine 2006
- **RNA interference**, gene silencing by double-stranded RNA
- An other key protein function

Andrew Z. **Fire**
(Stanford)

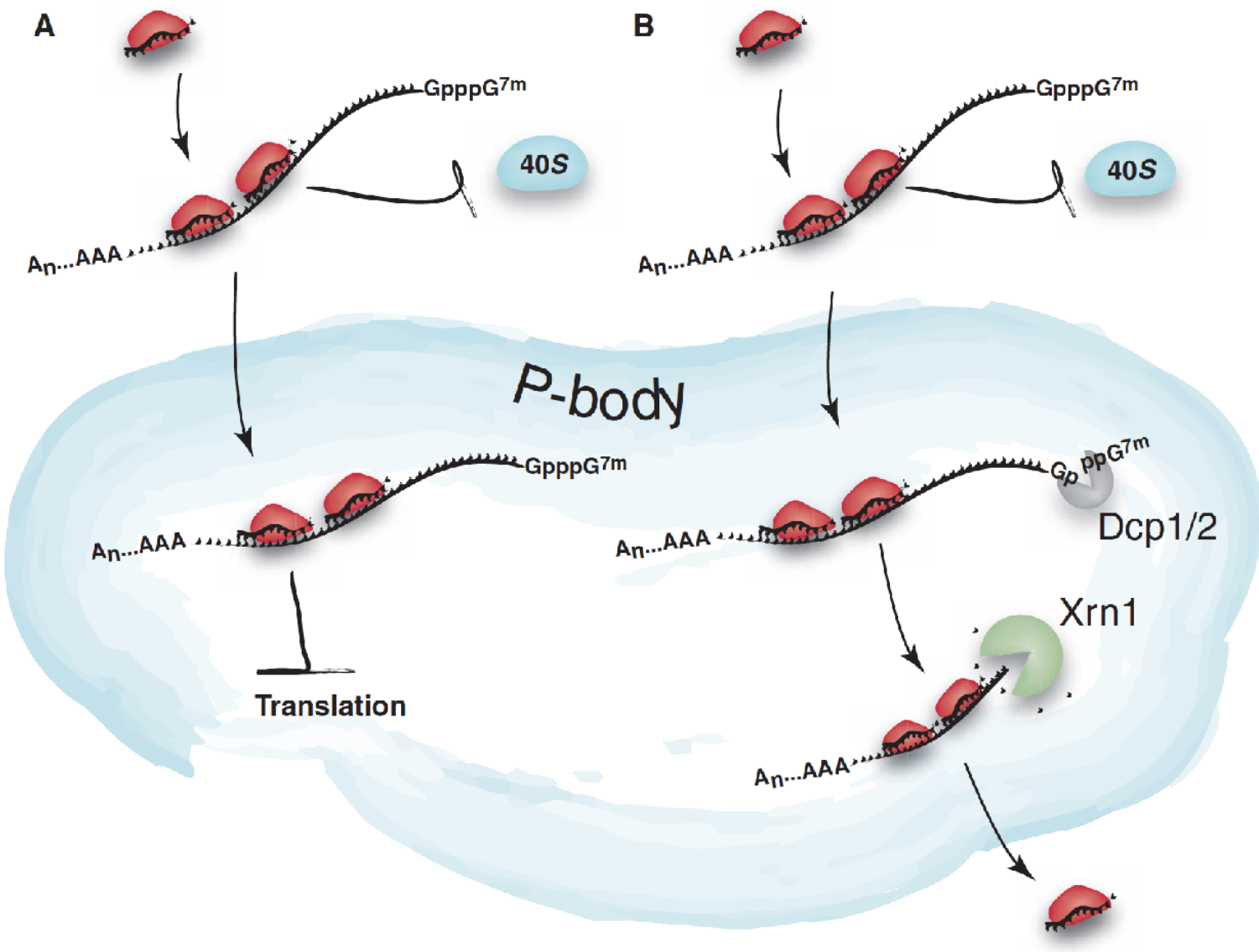
Craig C. **Mello**
(Massachusetts Medical School)

Central Dogma





Zamore and Haley. Ribo-gnome: the big world of small RNAs. Science (2005) vol. 309 (5740) pp. 1519-24



Zamore and Haley. Ribo-gnome: the big world of small RNAs. Science (2005) vol. 309 (5740) pp. 1519-24

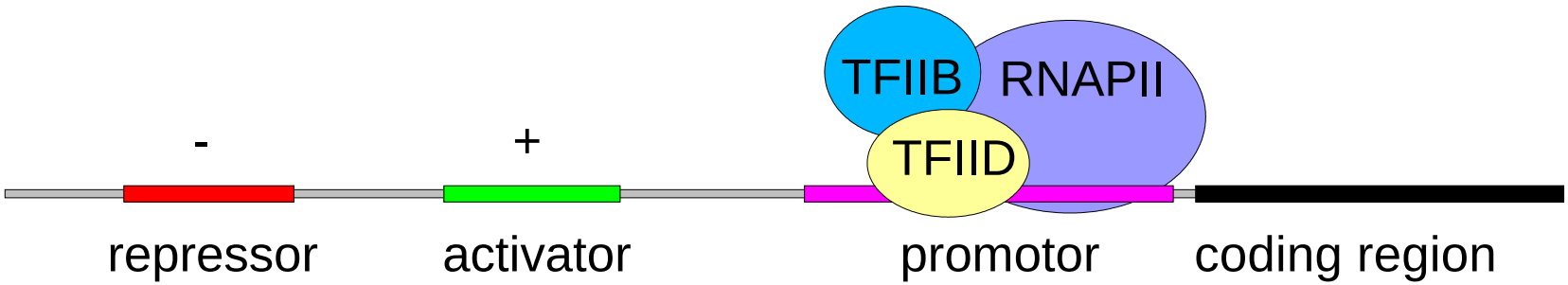
Understanding gene expression **regulation**

- The mechanisms modulating **gene expression** are numerous and complex
- However, there are two main control points; one for the **transcription** and the other for the **translation**

$$\text{DNA} \xrightarrow{\text{transcription}} \text{RNA} \xrightarrow{\text{translation}} \text{Protein}$$

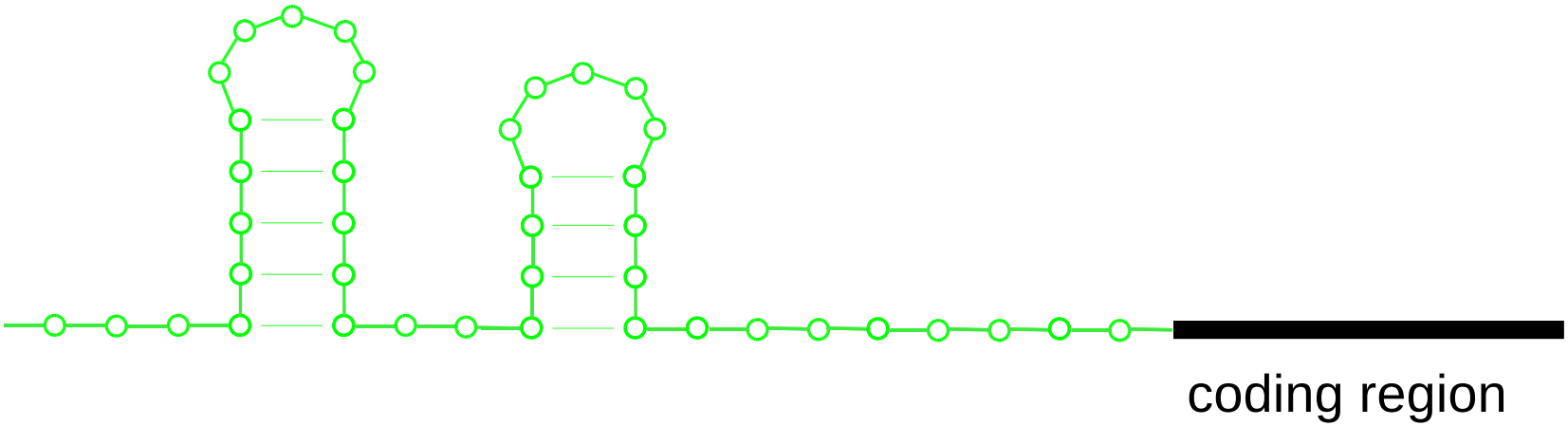
Step 1: Regulation of the **transcription**

The control of gene expression **at the level of the transcription** depends on regulatory elements that are situated upstream of the coding region



Those regulatory elements can be modelled with **regular languages** (they are sequence motifs)
Examples: $TATA[AT]A[AT]$ and $[CT][CT]CA[AG][AG]$

Step 2: **Post-transcriptional** regulation



It is becoming clear that **structural elements** are playing an important role modulating gene expression **at the level of the translation**

The Nobel Prize in Chemistry 2009

“for studies of the structure and function of the ribosome”

Venkatraman **Ramakrishnan**

MRC Laboratory of Molecular Biology

Cambridge, United Kingdom

Thomas A. **Steitz**

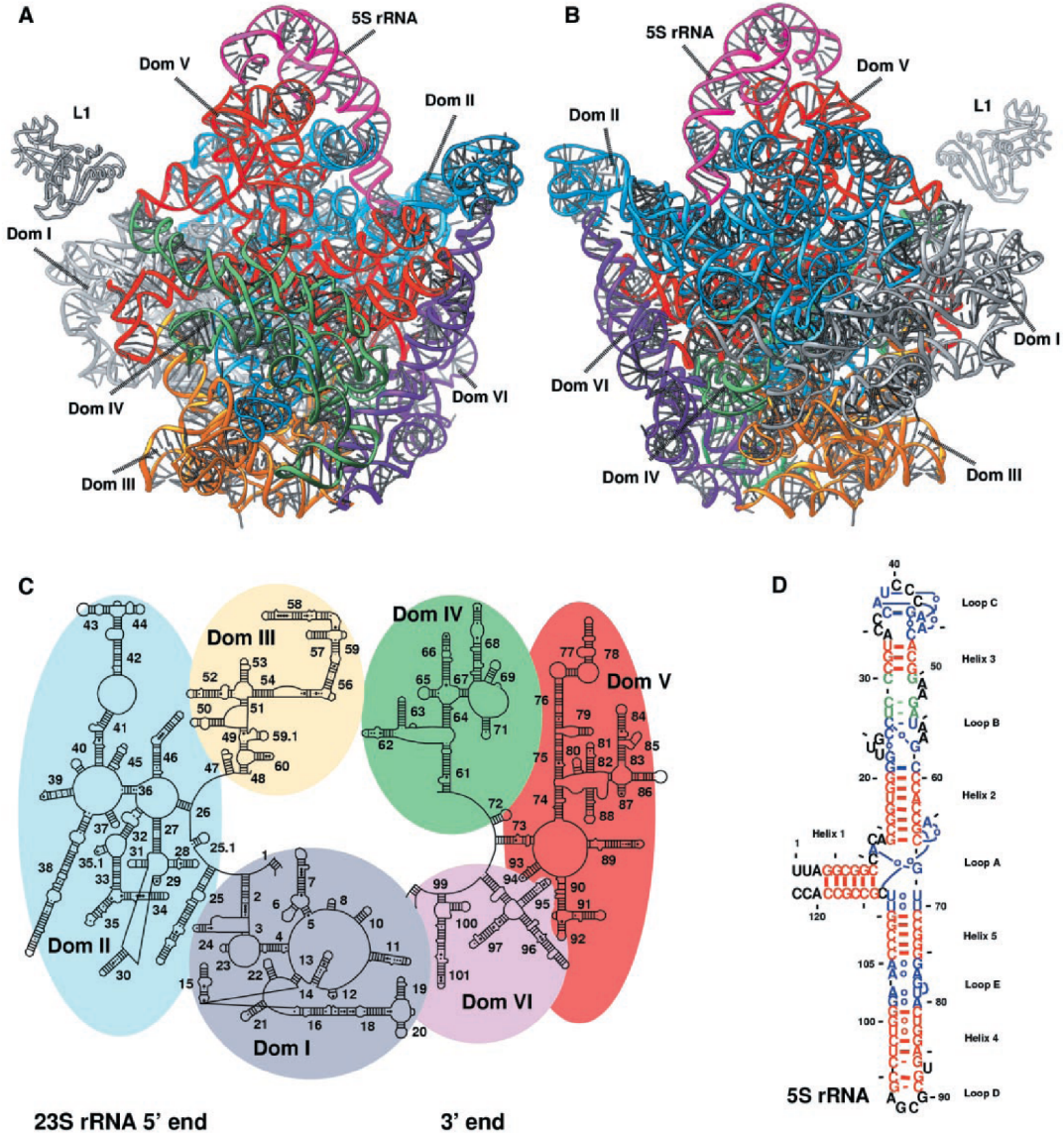
Yale University, Howard Hughes

Medical Institute New Haven, CT, USA

Ada E. **Yonath**

Weizmann Institute of Science

Rehovot, Israel



Ban et al. The complete atomic structure of the large ribosomal subunit at 2.4 A resolution. Science (2000) vol. 289 (5481) pp. 905-20

RNA and Protein Synthesis

mRNA: messager RNAs carries genetic information

tRNA: transfer RNAs are adapter molecules that recognize mRNA codons and carry a specific amino acid

rRNA: ribosomal RNAs account for $\frac{2}{3}$ of the molecular mass of the ribosome, which is a large RNA+protein complex responsible for translating genomic information (stored in mRNAs) into proteins

Cech, T. R., & Steitz, J. A. (2014). The noncoding RNA revolution-trashing old rules to forge new ones. **Cell**, **157**(1), 77–94. `http://doi.org/10.1016/j.cell.2014.03.008`

Non-coding RNAs

miRNA: microRNAs modulate the development in C. elegans, Drosophila, and mammals ($\sim$20 nt)

snRNA: small nuclear RNAs are involved in splicing of eukaryotic mRNAs ($\sim$200 nt)

snoRNA: small nucleolar RNA direct nucleotide modifications in rRNAs ($\sim$100 nt)

gRNA: guide RNAs play an important role in editing of certain mRNAs in trypanosomes ($\sim$ 70 nt)

Non-Coding RNAs (contd)

tmRNA: have the combined features of tRNAs and mRNAs and plays a role in translation regulation in bacterial genomes ($\sim$400 nt)

SRP: (signal recognition particle RNA-protein complex) directs newly synthesized proteins through the endoplasmic reticulum

M1 RNA: is the catalytic part of Ribonuclease P in bacteria, involves in the maturation of pre-tRNA ($\sim$375 nt)

TERC: telomerase RNA is an integral part of telomerase enzyme that serves as a template for the synthesis of the telomeres ($\sim$450 nt)

...

Rfam database

- **Rfam** 14 (August 2018) contains **2,4791** RNA families
- For each family
  - Multiple sequence alignment (seed, full)
  - Consensus secondary structure (from literature or predicted)
  - Covariance model
- rfam.org
- Kalvari, I. et al. Non-Coding RNA Analysis Using the Rfam Database. *Curr Protoc Bioinformatics* **62**, e51 (2018).
- Kalvari, I. et al. Rfam 13.0: shifting to a genome-centric resource for non-coding RNA families. *Nucleic Acids Res* **46**, D335–D342 (2018).

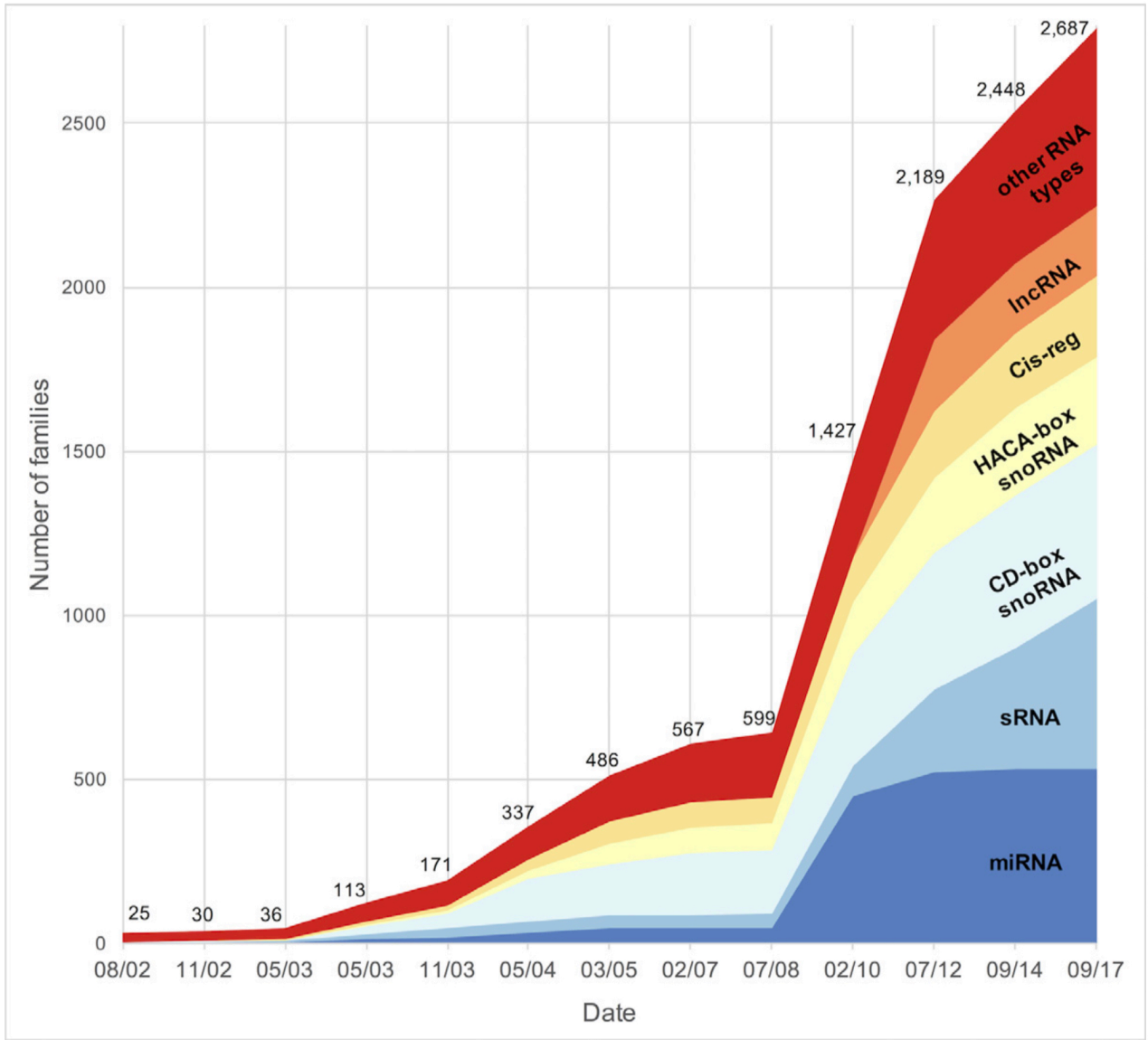
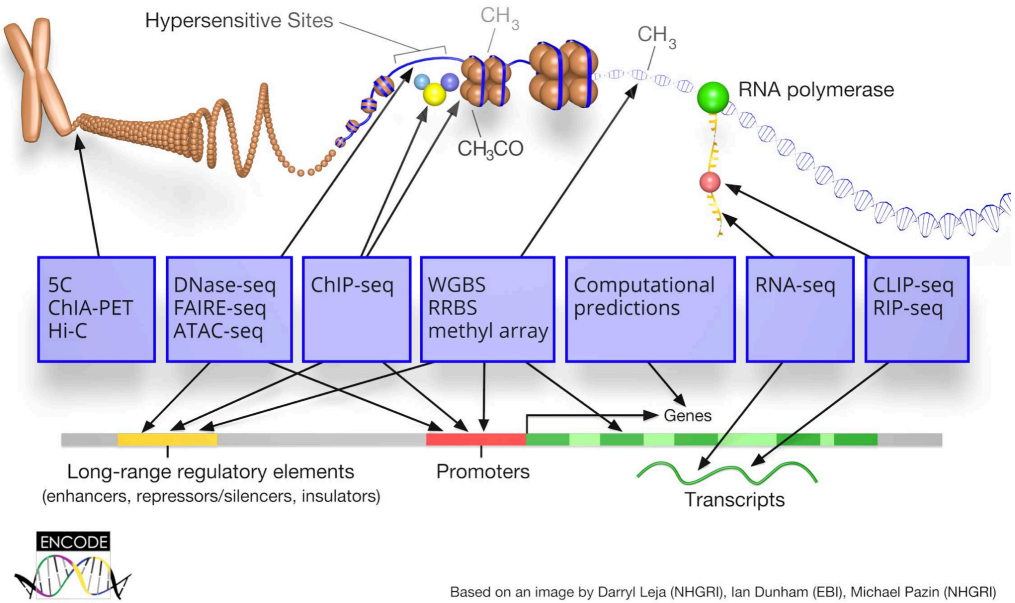


Figure 1. Growth in the number of RNA families grouped by RNA type in major database releases. The *other RNA types* group includes types with less than 50 families, such as rRNA, tRNA, snRNA or riboswitches.

RNACentral — rnacentral.org

- The RNAcentral Consortium. (2017). RNAcentral: a comprehensive database of non-coding RNA sequences. *Nucleic Acids Research*, **45**(D1), D128–D134. http://doi.org/10.1093/nar/gkw1008
- Petrov, A. I., Kay, S. J. E., Gibson, R., Kulesha, E., Staines, D., Bruford, E. A., et al. (2015). RNAcentral: An international database of ncRNA sequences. *Nucleic Acids Research*, **43**(D1), D123–D129. http://doi.org/10.1093l/narlgku991
- Bateman, A., Agrawal, S., Birney, E., Bruford, E. A., Bujnicki, J. M., Cochrane, G., et al. (2011). RNAcentral: A vision for an international database of RNA sequences. *RNA*, **17**(11), 1941–1946. http://doi.org/10.1261/rna.2750811

ENCODE: Encyclopedia of DNA Elements



Based on an image by Darryl Leja (NHGRI), Ian Dunham (EBI), Michael Pazin (NHGRI)

About ENCODE Project | Getting Started | Experiments

Search ENCODE portal

ENCODE

About ENCODE Encyclopedia | Candidate Regulatory Elements

Search for Candidate Regulatory Elements

Hosted by SCREEN

Human hg19 | Mouse mm10

ENCODE (ENCyclopedia Of DNA Elements)

- "The human genome is pervasively transcribed, such that **the majority of its bases are associated with at least one primary transcript** (…)"

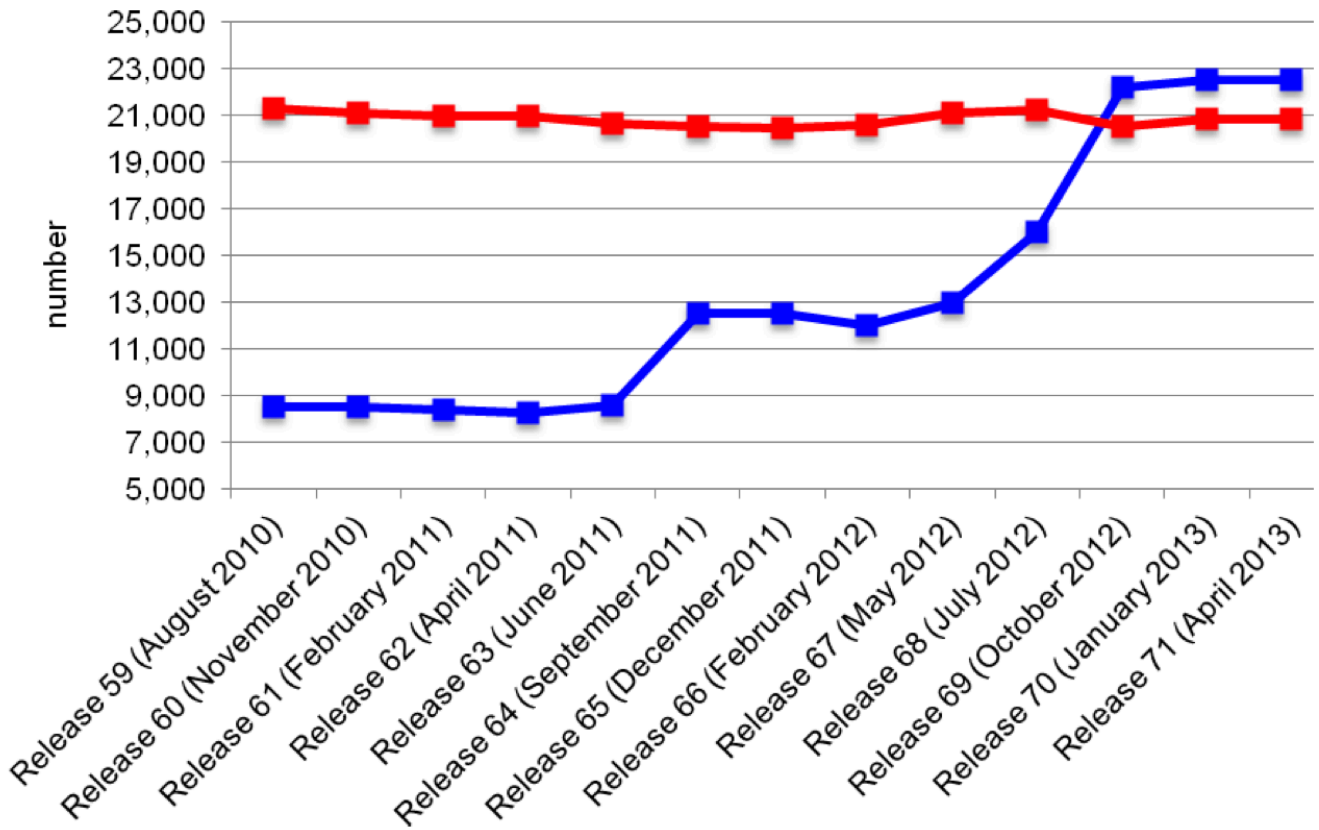
Birney et al. *Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project.* Nature (2007) vol. **447** (7146) pp. 799-816

How Many **Non-Coding RNAs**?

- **48,479** candidates in the human genome (EvoFold)
  Pedersen et al. Identification and classification of conserved RNA secondary structures in the human genome. PLoS Comput Biol (2006) vol. 2 (4) pp. e33
- Studies based on the ENCODE data set
  - **3,267** RNAz, **3,134** EvolFold
    Washietl et al. Structured RNAs in the ENCODE selected regions of the human genome. Genome Res (2007) vol. 17 (6) pp. 852-64
  - **4,933** CMfinder
    Torarinsson et al. Comparative genomics beyond sequence-based alignments: RNA structures in the ENCODE regions. Genome Res (2008) vol. 18 (2) pp. 242-51

Protein versus **ncRNA** annotations

Figure 4. Number of non-coding and protein-coding genes annotated over the last Ensembl releases. The *x*-axis indicates the number and the date of the release. The vertical axis reports the number of ncRNA (blue line) and protein-coding genes (red line).



- Bussotti,G. et al. (2013) Detecting and comparing non-coding RNAs in the high-throughput era. *Int J Mol Sci,* 14, 15423-15458.

Action **Mechanisms**

- direct **base-pairing** with RNA or DNA target: snoRNAs, miRNAs
- mimic the **structure** of other nucleic acids (or proteins?): tmRNA, some snRNAs, IRES
- **catalyst**: RNAs P

John S. **Mattick**

- Over 430 publications, **57,330 citations**!
- Over **150 co-authors**, h-index $=$ 111 (Google Scholar)
- **Garvan Institute of Medical Research**, Australia, Sydney

John S. Mattick (contd)

- Morris,K.V. and Mattick,J.S. (2014) **The rise of regulatory RNA**. Nat Rev Genet, 15, 423–437.

"it seems that RNA is the **computational engine** of cell biology, developmental biology, brain function and perhaps even evolution itself. The complexity and interconnectedness of these systems should not be cause for concern but rather the motivation for **exploring the vast unknown universe of RNA regulation, without which we will not understand biology**."

Smith et al. (2013) **Widespread purifying selection on RNA structure in mammals**. Nucleic Acids Res, 41, 8220-8236.
Amaral,P.P. et al. (2008) **The eukaryotic genome as an RNA machine**. Science, 319, 1787–1789.
Mattick et al. **Non-coding RNA**. Hum Mol Genet (2006) vol. 15 Spec No 1 pp. R17-29
Carninci et al. **The transcriptional landscape of the mammalian genome**. Science (2005) vol. 309 (5740) pp. 1559-63
Mattick. **RNA regulation: a new genetics?**. Nat Rev Genet (2004) vol. 5 (4) pp. 316-23
Mattick. **Challenging the dogma: the hidden layer of non-protein-coding RNAs in complex organisms.** Bioessays (2003) vol. 25 (10) pp. 930-9
Mattick et al. **The evolution of controlled multitasked gene networks**: the role of introns and other noncoding RNAs in the development of complex organisms. Mol Biol Evol (2001) vol. 18 (9) pp. 1611-30

Fascinating RNAs

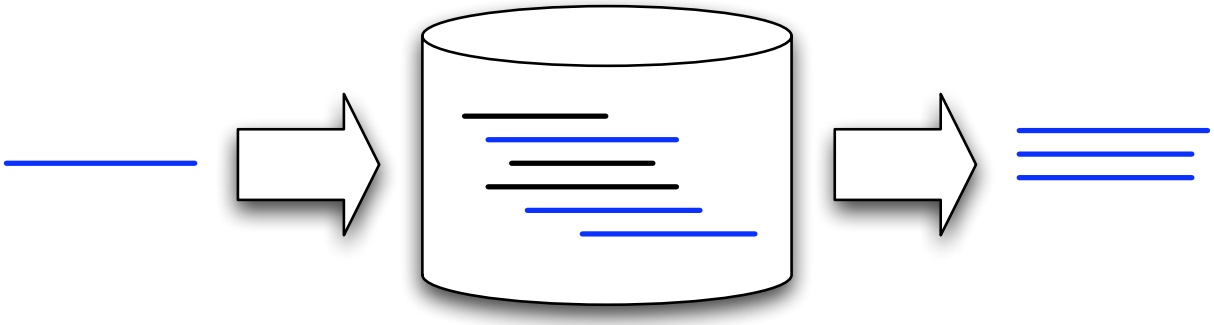
- **Versatile** molecules that can **carry information**, as DNA does, and perform **catalytic functions**, as proteins do

Fascinating RNAs

- **Versatile** molecules that can **carry information**, as DNA does, and perform **catalytic functions**, as proteins do
- Seem to be governed by simpler laws, as a result RNA analysis is a **big bioinformatics success** (see Gutell's work on predicting secondary and tertiary interactions, and Major's work on predicting tertiary structure)

Database Search Problem

Find all GenBank gene's that are similar to *Clostridium botulinum*'s toxin



```
>gi|27867582(fragment of the known Clostridium botuninum toxin gene)
GTGAATCAGCACCTGGACTTTCAGATGAAAAATTAAATTTAACTATCCAAAATGATGCTTATATACCAAAATATGATTCTAATGGAACAA
GTGATATAGAACAACATGATGTTAATGAACTTAATGTATTTTTCTATTTAGATGCACAGAAAGTGCCCGAAGGTGAAAATAATGTCAATC
TCACCTCTTCAATTGATACAGCATTATTAGAACAACCTAAAATATATACATTTTTTTCATCAGAATTTATTAATAATGTCAATAAACCTG
TGCAAGCAGC
```

Database Search Problem

```
>gi|49138|emb|X68262.1|CBBONTF  C.barati gene for type F neurotoxin

Length=4073 Score = 81.8 bits (41),  Expect = 1e-12
Identities = 99/121 (82.82%), Gaps = 2/121 (0.02%)
Strand=Plus/Plus

Query  48    CAAAATGATGCTTATATACCAAAATATGATTCTAATGGAACAAGTGATATAGAACAACAT  107
             ||||||||| ||||  | ||||||||||||||||||| |||||||| ||| |  || ||
Sbjct  1712  CAAAATGATTCTTACGTTCCAAAATATGATTCTAATGGTACAAGTGAAATAAA-GAATAT  1771

Query  108   GATGTTAATGAACTTAATGTATTTTTCTATTTAGATGCACAGAAAGTGCC-GAAGGTGAA  167
               |||| || |||| |||||||||||||||||| ||||||| ||||  || |||||||||
Sbjct  1772  ACTGTTGATAAACTAAATGTATTTTTCTATTTATATGCACAAAAAGCTCCTGAAGGTGAA  1831

Query  168   A  168             |
Sbjct  1832  A  1832
...
```

Pairwise **Sequence Alignment**

Pairwise **Sequence Alignment**

- Positions along the sequence are **independent** and **identically distributed** *i.i.d.*

Pairwise **Sequence Alignment**

- Positions along the sequence are **independent** and **identically distributed** *i.i.d.*
- **Independence** is necessary for the development of efficient exact (Smith-Waterman) or heuristics (such as BLAST) algorithms

Pairwise **Sequence Alignment**

- Positions along the sequence are **independent** and **identically distributed** *i.i.d.*
- **Independence** is necessary for the development of efficient exact (Smith-Waterman) or heuristics (such as BLAST) algorithms
- The execution time of the exact algorithms grows proportionally to the product of the **size of the database** times the **size of the input sequence**

RNA Sequence Alignment (Toy Example)

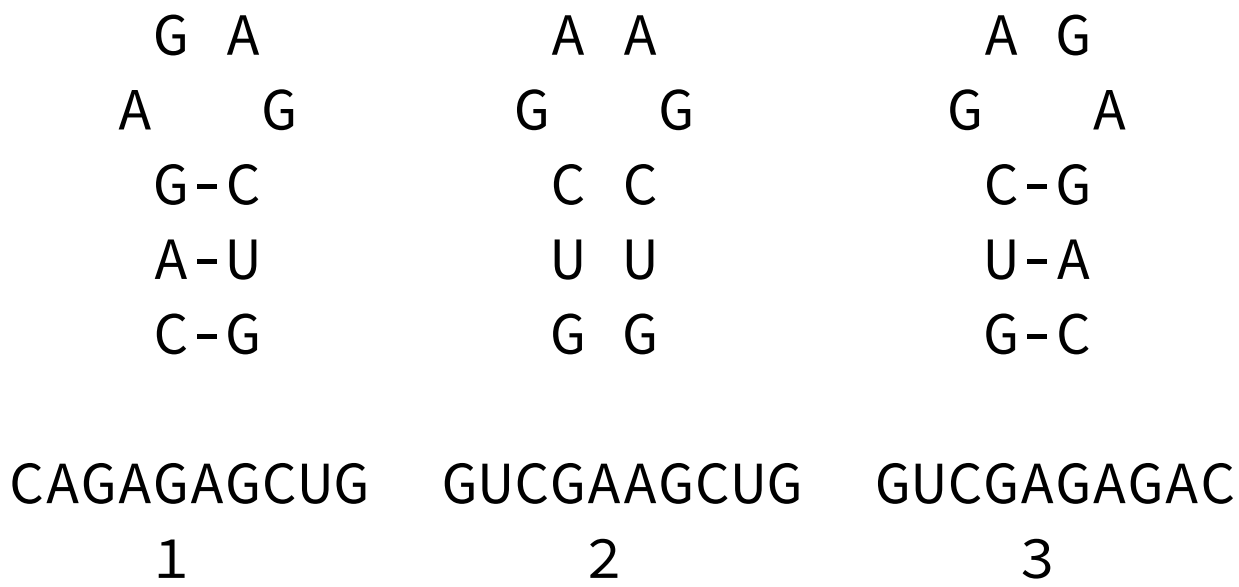
```
1   GUCGAGAGAC
    !!!!!
2   GUCGAAGCUG
          !!!!!
3   CAGAGAGCUG
```

RNA Sequence Alignment (Toy Example)

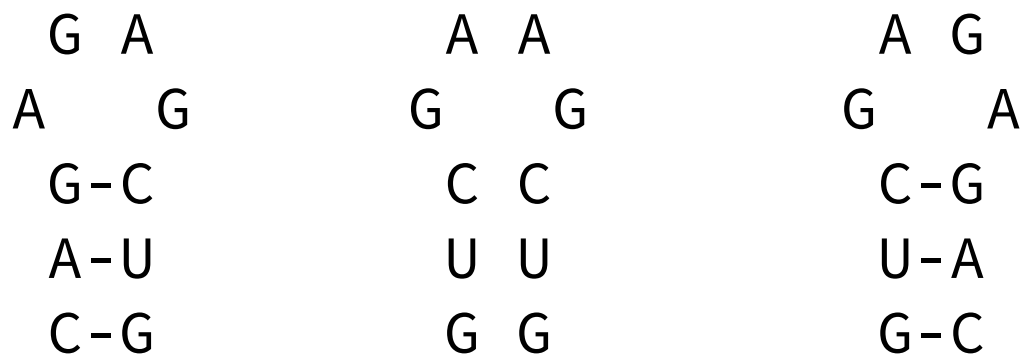
```
1   GUCGAGAGAC
    ! ! ! ! !
2   GUCGAAGCUG
          ! ! ! ! !
3   CAGAGAGCUG
```

1 and 2 are 50% identical (similarly for 2 and 3), however, 1 and 3 don't seem to have anything in common

RNA Sequence Alignment (Toy Example)



RNA Sequence Alignment (Toy Example)

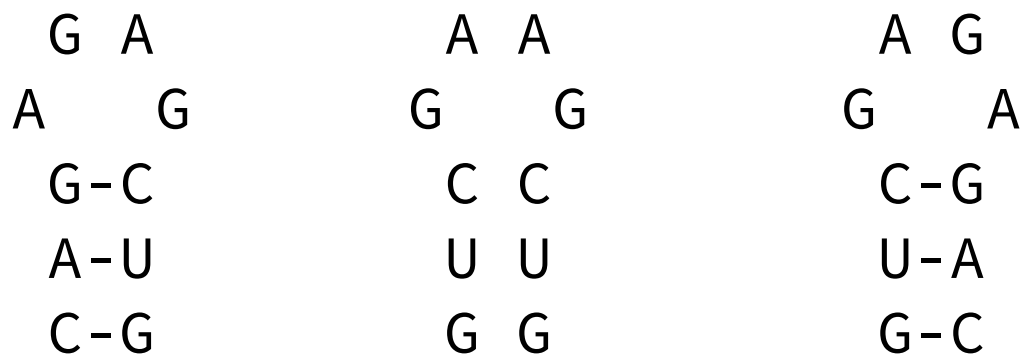


```
  G  A          A  A          A  G
A    G        G    G        G    A
  G-C          C  C          C-G
  A-U          U  U          U-A
  C-G          G  G          G-C
```

```
CAGAGAGCUG   GUCGAAGCUG   GUCGAGAGAC
     1            2            3
```

Yes, but sequences 1 and 3 share the same secondary structure!

RNA Sequence Alignment (Toy Example)

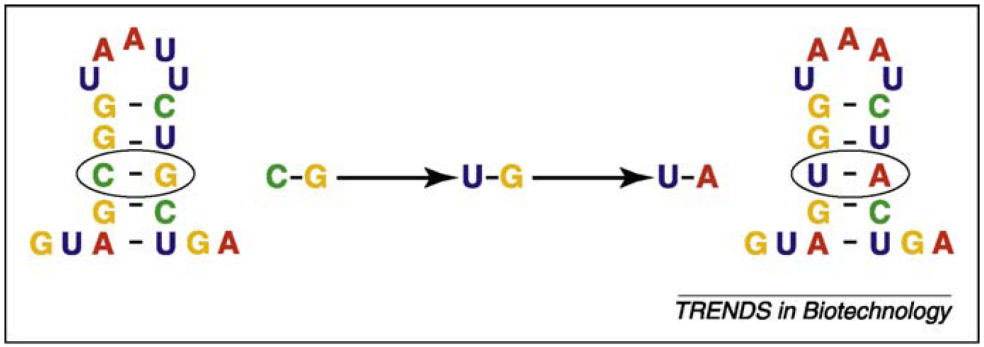


CAGAGAGCUG GUCGAAGCUG GUCGAGAGAC

1 2 3

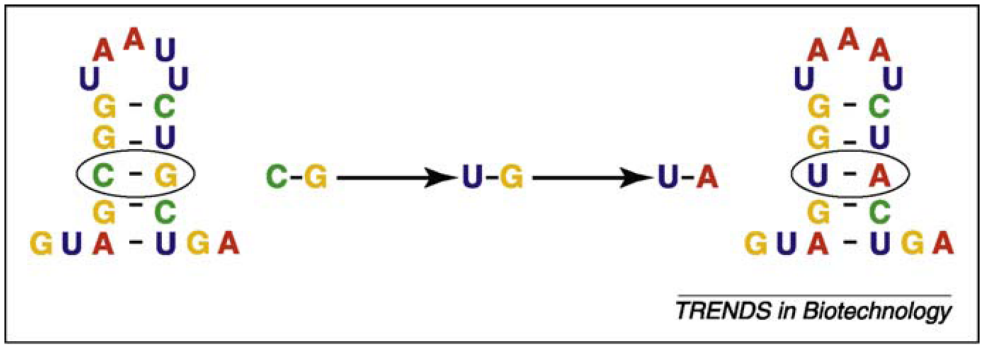
Yes, but sequences 1 and 3 share the same secondary structure!
Yet, sequences 1 and 3 cannot be aligned!

Caveat



- RNAs conserve **secondary structure** interactions more than they conserve their sequence

Caveat



- RNAs conserve **secondary structure** interactions more than they conserve their sequence
- **Traditional bioinformatics tools**, assuming that positions are independent, perform poorly

Given an **RNA sequence** $S = s_1, s_2 \ldots s_n$, where s_i is the ith nucleotide. A **secondary structure** is an ordered list of pairs, $i.j$, $1 \leq i < j \leq n$ such that:

Given an **RNA sequence** $S = s_1, s_2 \ldots s_n$, where s_i is the ith nucleotide. A **secondary structure** is an ordered list of pairs, $i.j$, $1 \leq i < j \leq n$ such that:

- $j - i \geq c$, where $c = 4$ for instance

Given an **RNA sequence** $S = s_1, s_2 \ldots s_n$, where s_i is the ith nucleotide. A **secondary structure** is an ordered list of pairs, $i.j$, $1 \leq i < j \leq n$ such that:

- $j - i \geq c$, where $c = 4$ for instance
- Given $i.j$ and $i'.j'$, two base pairs, then either:

Given an **RNA sequence** $S = s_1, s_2 \ldots s_n$, where s_i is the ith nucleotide. A **secondary structure** is an ordered list of pairs, $i.j$, $1 \leq i < j \leq n$ such that:

- $j - i \geq c$, where $c = 4$ for instance
- Given $i.j$ and $i'.j'$, two base pairs, then either:
  - $i = i'$ and $j = j'$ (they are the same)

Given an **RNA sequence** $S = s_1, s_2 \ldots s_n$, where s_i is the ith nucleotide. A **secondary structure** is an ordered list of pairs, $i.j$, $1 \leq i < j \leq n$ such that:

- $j - i \geq c$, where $c = 4$ for instance
- Given $i.j$ and $i'.j'$, two base pairs, then either:
  - $i = i'$ and $j = j'$ (they are the same)
  - $i < j < i' < j'$ ($i.j$ precedes $i'.j'$)

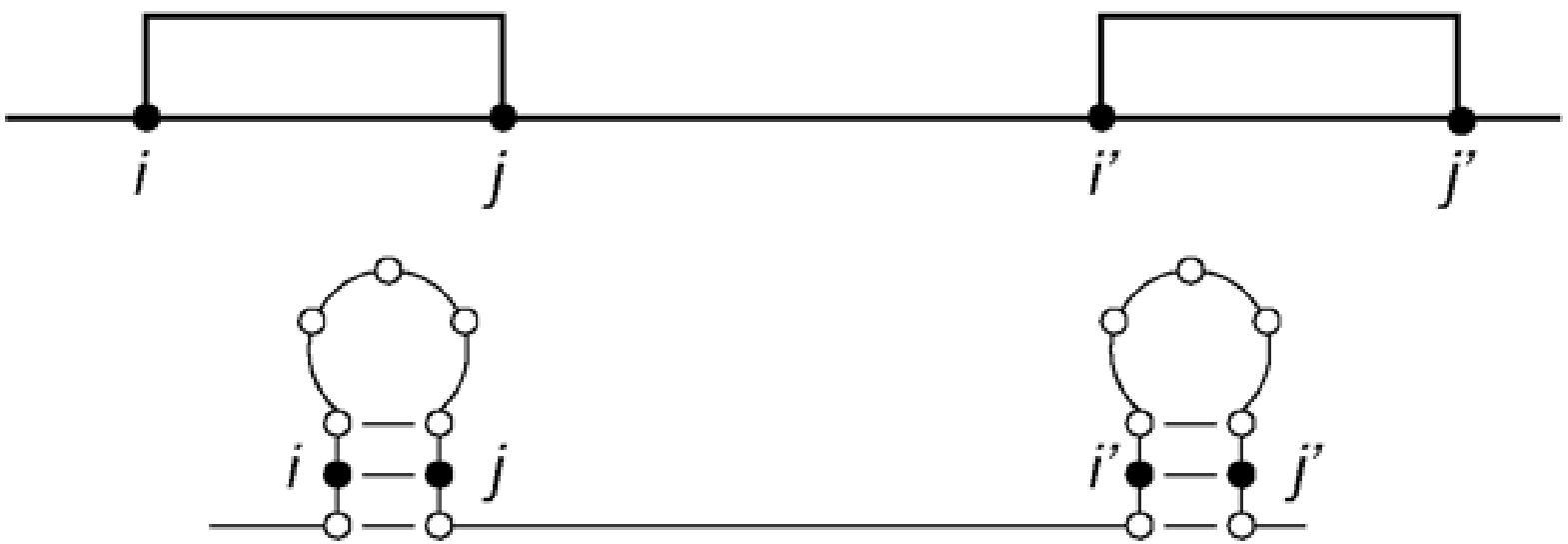
Given an **RNA sequence** $S = s_1, s_2 \ldots s_n$, where s_i is the ith nucleotide. A **secondary structure** is an ordered list of pairs, $i.j$, $1 \leq i < j \leq n$ such that:

- $j - i \geq c$, where $c = 4$ for instance
- Given $i.j$ and $i'.j'$, two base pairs, then either:
  - $i = i'$ and $j = j'$ (they are the same)
  - $i < j < i' < j'$ ($i.j$ precedes $i'.j'$)
  - $i < i' < j' < j$ ($i.j$ includes $i'.j'$)

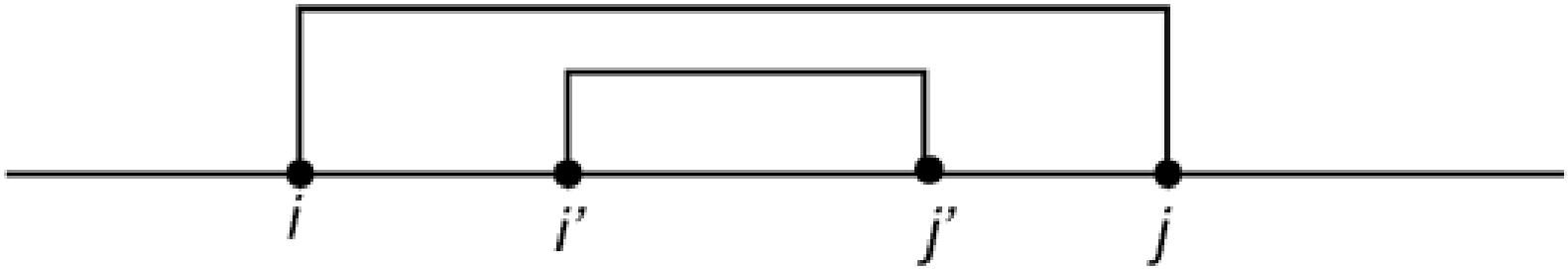
Given an **RNA sequence** $S = s_1, s_2 \ldots s_n$, where s_i is the ith nucleotide. A **secondary structure** is an ordered list of pairs, $i.j$, $1 \leq i < j \leq n$ such that:

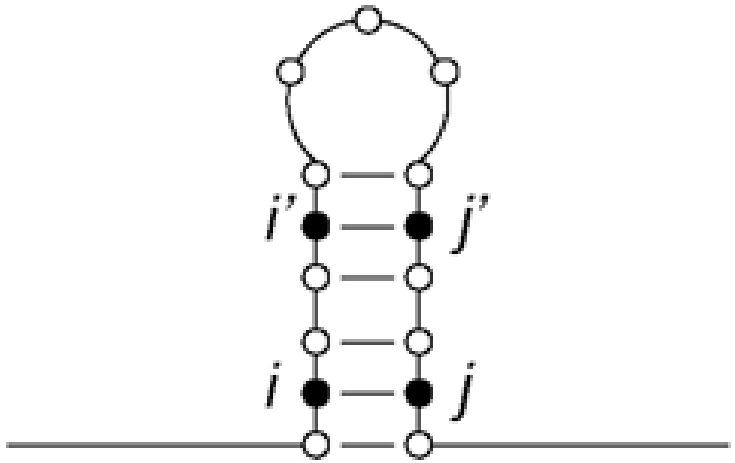
- $j - i \geq c$, where $c = 4$ for instance
- Given $i.j$ and $i'.j'$, two base pairs, then either:
  - $i = i'$ and $j = j'$ (they are the same)
  - $i < j < i' < j'$ ($i.j$ precedes $i'.j'$)
  - $i < i' < j' < j$ ($i.j$ includes $i'.j'$)
  - $i < i' < j < j'$ (pseudoknot)

$i < j < i' < j'$ ($i.j$ precedes $i'.j'$)

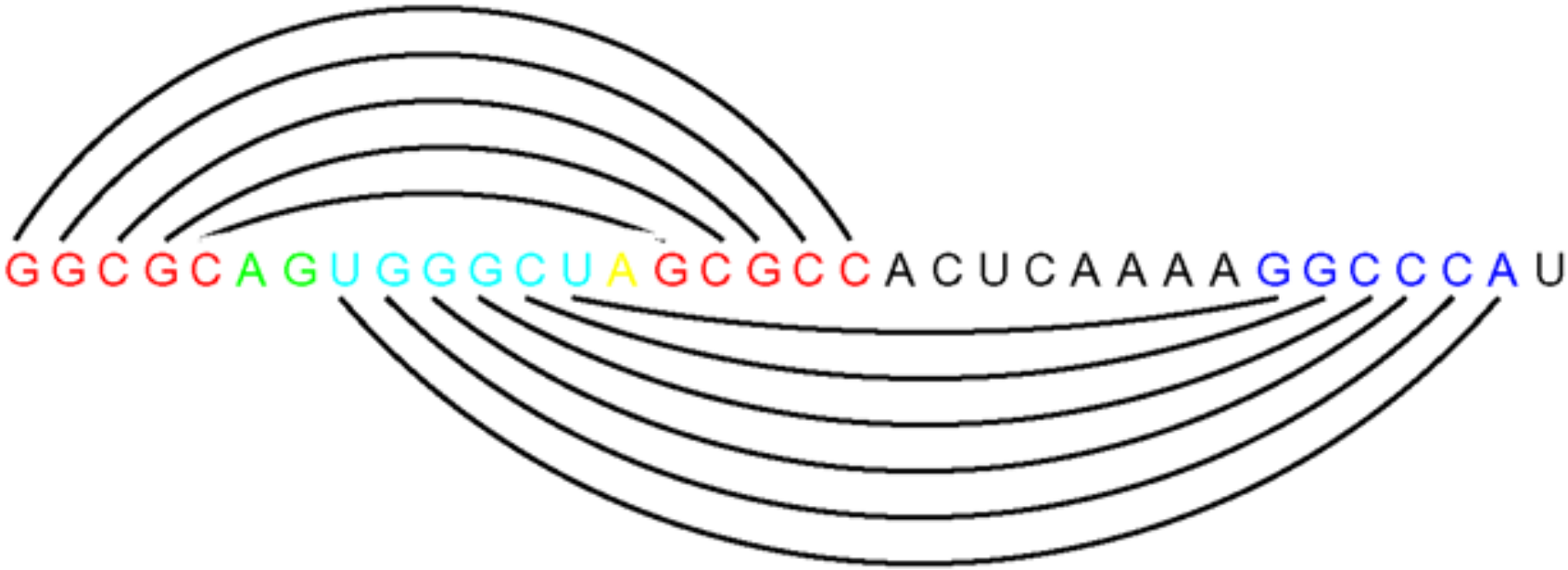


$i < i' < j' < j$ ($i.j$ includes $i'.j'$)

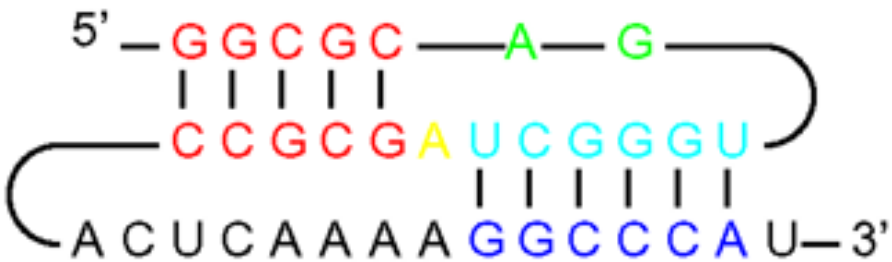




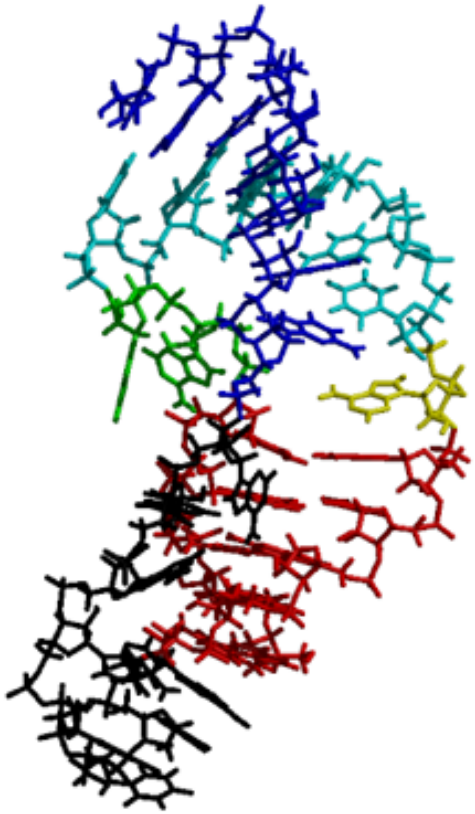
Pseudo-knotted structure (**pseudoknots**)



Pseudo-knotted structure (**pseudoknots**)



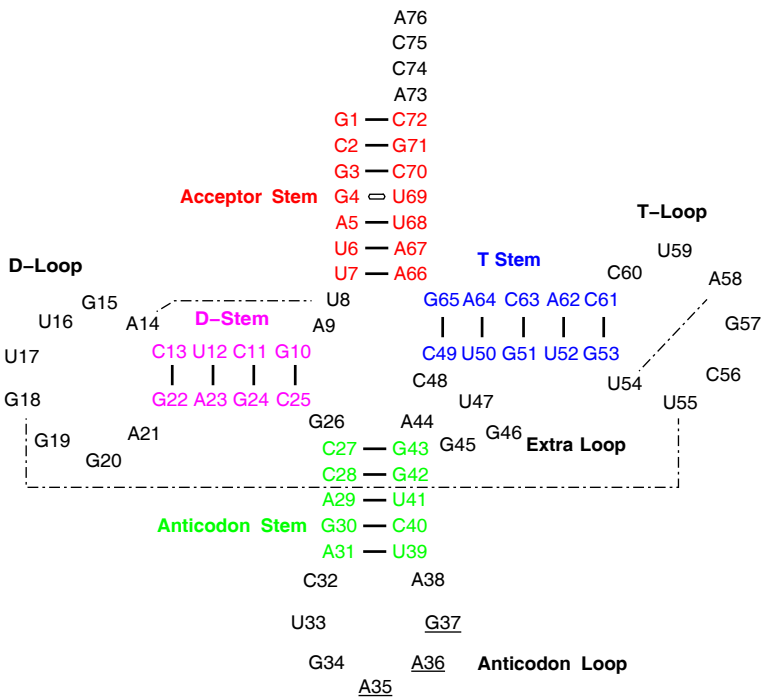
Pseudo-knotted structure (**pseudoknots**)

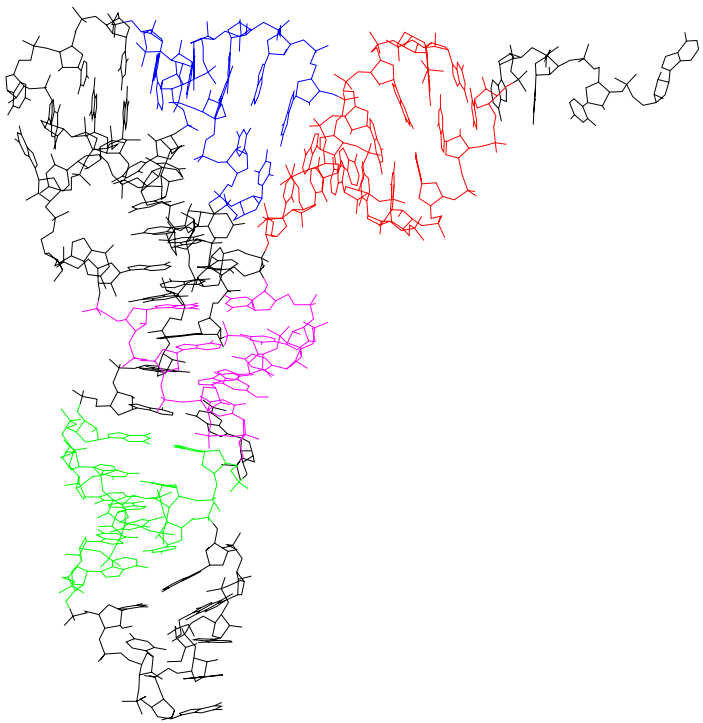


RNA 1, 2, 3

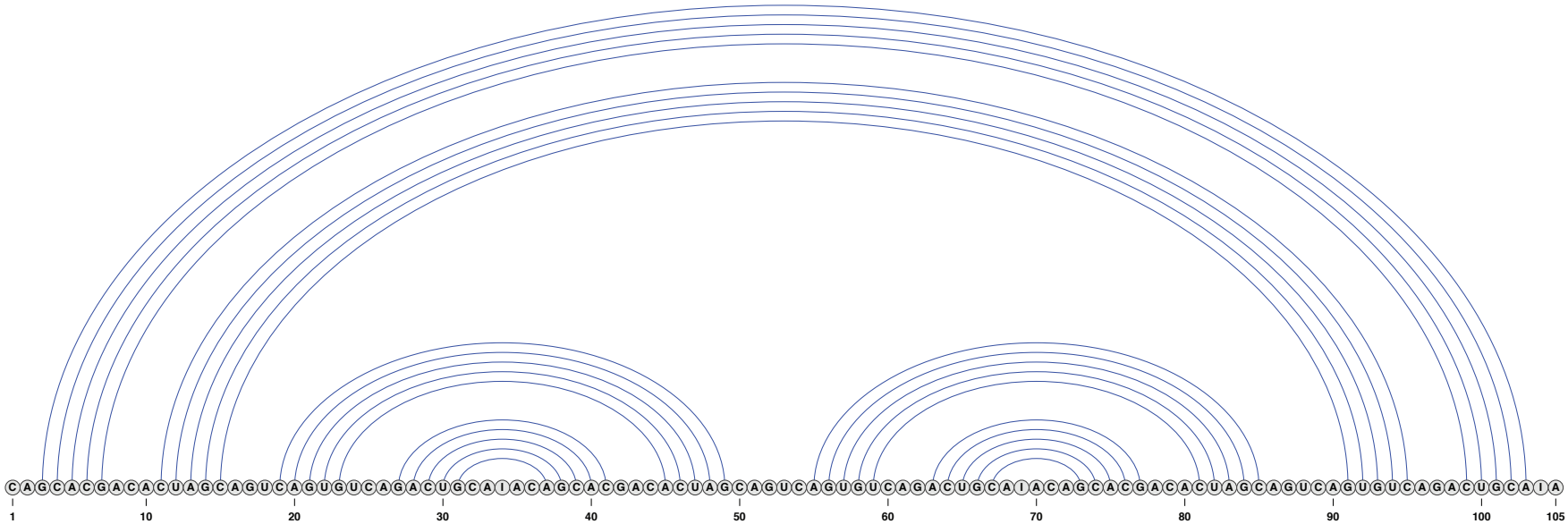
GCGGAUUUAGCUCAGUUGGGAGAGCGCCAGACUGAAGAUCUGGAGGUCCUGUGUUCGAUCCACAGAAUUCGCACCA

1 10 20 30 40 50 60 70



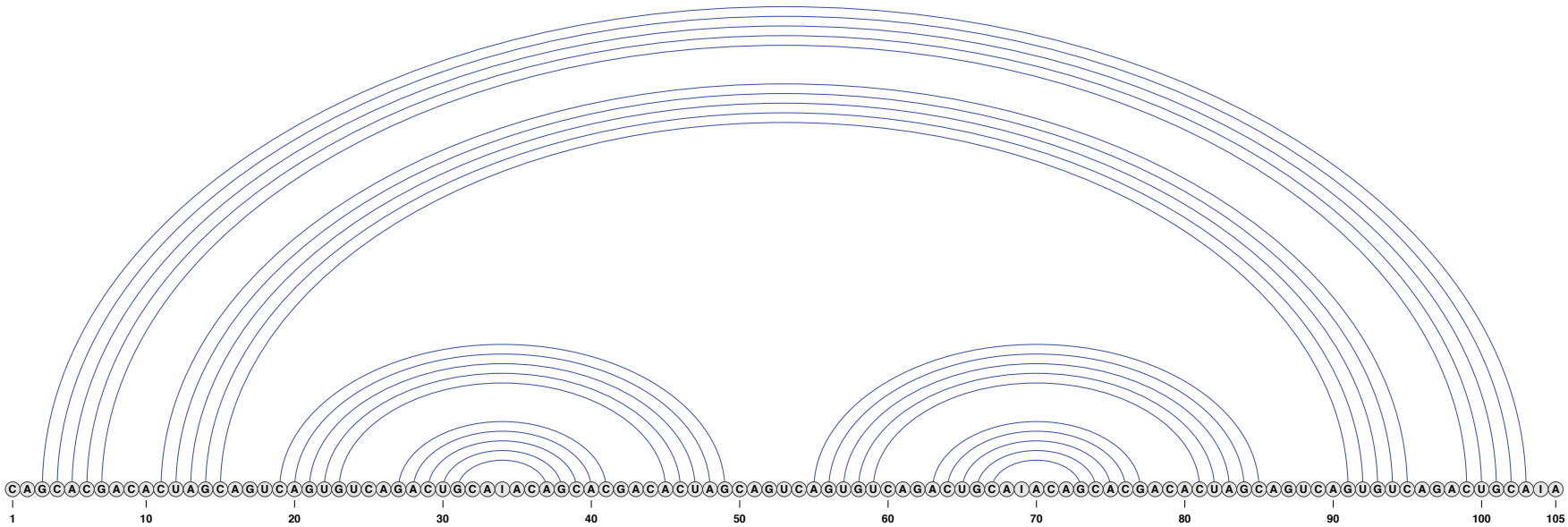


Representation: **arcs**



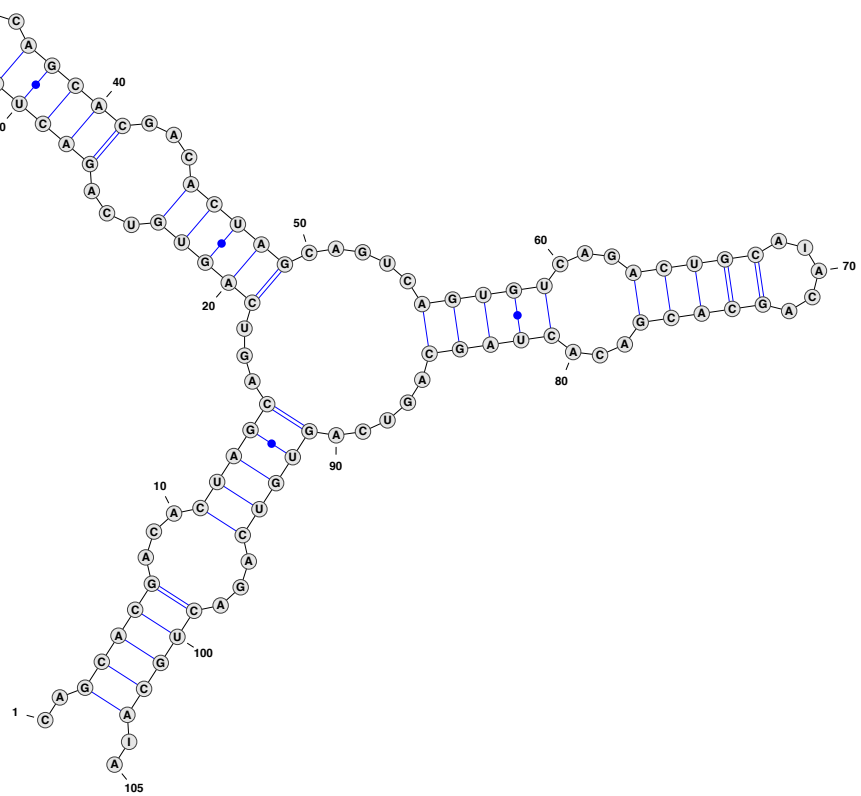
GCACGACACUAGCAGUCAGUGUCAGACUGCAIACAGCACGACACUAGCAGUCAGUGUCAGACUGCAIACAGCACGACACUAGCAGUCAGUGUC

Representation: **brackets notation**



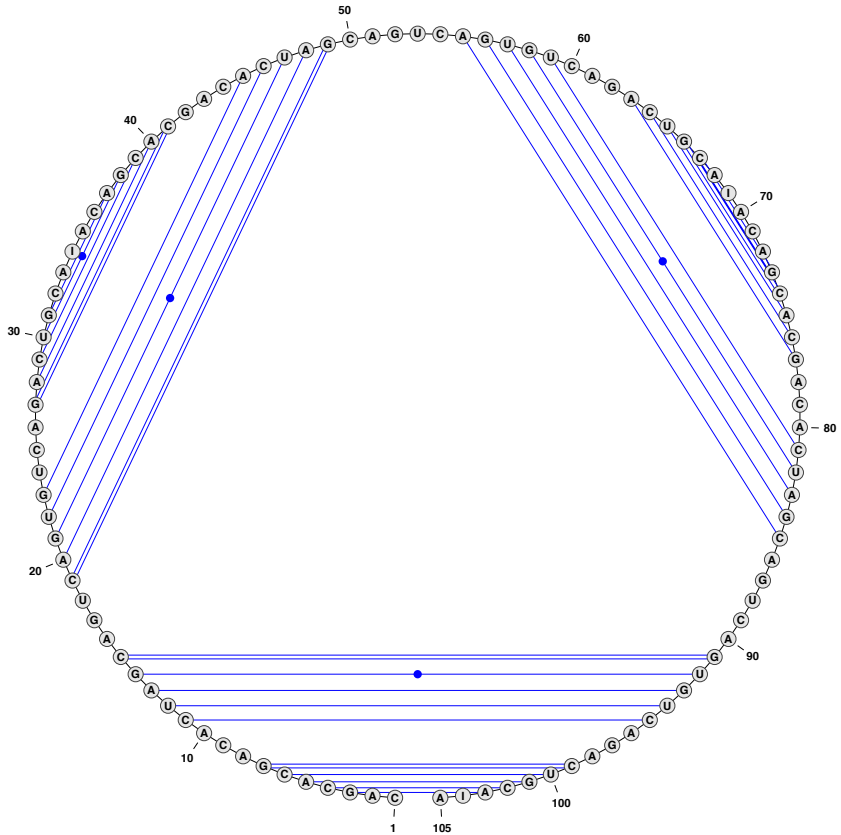

```
GCACGACACUAGCAGUCAGUGUCAGACUGCAIACAGCACGACACUAGCAGUCAGUGUCAGACUGCAIACAGCACGACACUAGCAGUCAGUGUC
((((((...((((((...((((((...((((((.....))))))...))))))..((((((...((((((.....))))))...))))))....))))))
```

Representation: **brackets notation**



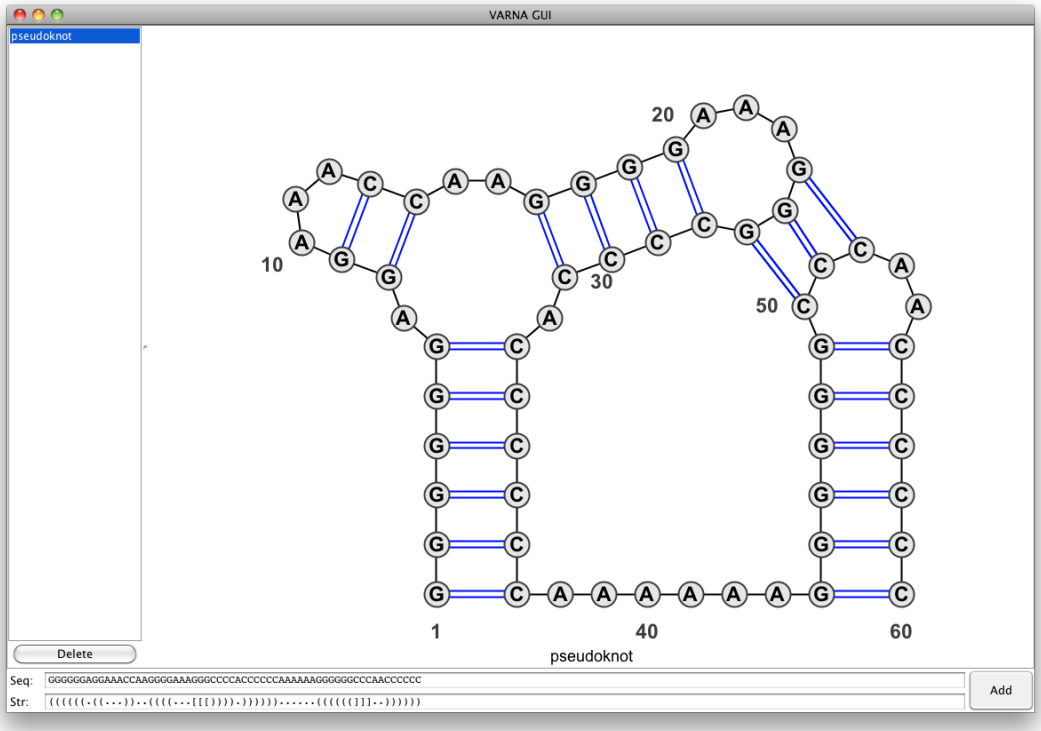
```
GCACGACACUAGCAGUCAGUGUCAGACUGCAIACAGCACGACACUAGCAGUCAGUGUCAGACUGCAIACAGCACGACACUAGCAGUCAGUGUC
(((((...(((((...(((((...(((((.....)))))...)))))).....(((((...(((((.....)))))...)))))).....)))))
```

Representation: **circular**




```
GCACGACACUAGCAGUCAGUGUCAGACUGCAIACAGCACGACACUAGCAGUCAGUGUCAGACUGCAIACAGCACGACACUAGCAGUCAGUGUC
(((((...(((((...(((((...(((((.....)))))...)))))).....(((((...(((((.....)))))...)))))).....)))))
```

VARNA



VARNA: Interactive drawing and editing of the RNA secondary structure Kévin Darty, Alain Denise and Yann Ponty

Bioinformatics, pp. 1974-1975, Vol. 25, no. 15, 2009

Ribsome

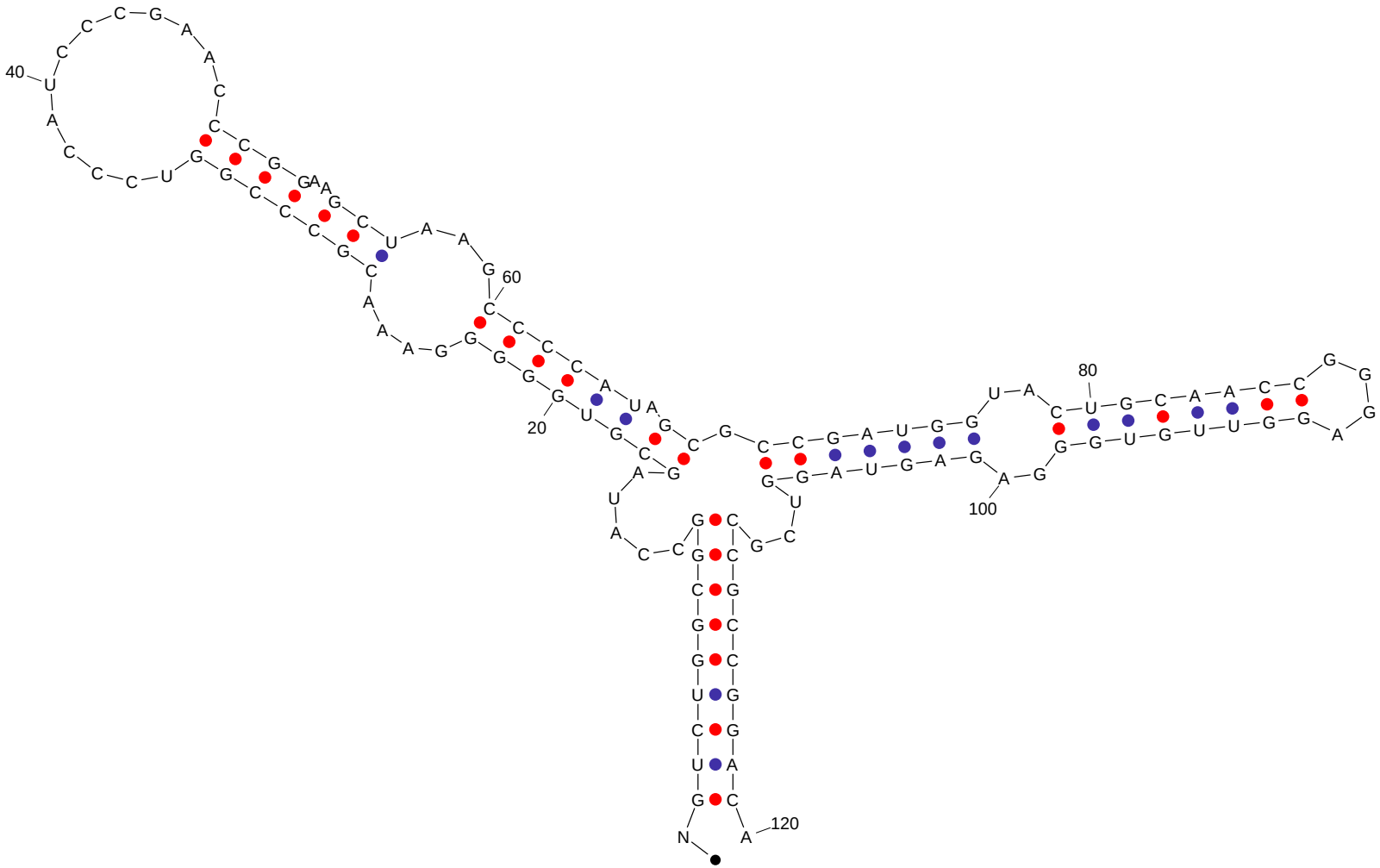
- Large ribo-protein complex responsible for protein translation
- $\frac{2}{3}$ nucleic acid, $\frac{1}{3}$ protein
- In **Eukaryotes**, the ribosomes are designated as 80S*
- 80S has two subunits: small (40S) and large (60S)
- **Small subunit** consists of 18S (1900 nucleotides) + 33 proteins
- **Large subunit** consists of 5S (120 nucleotides), 28S (4700 nucleotides), 5.8S (160 nucleotides) + 49 proteins

*The unit of measurement is the Svedberg unit
a measure of the rate of sedimentation in centrifugation

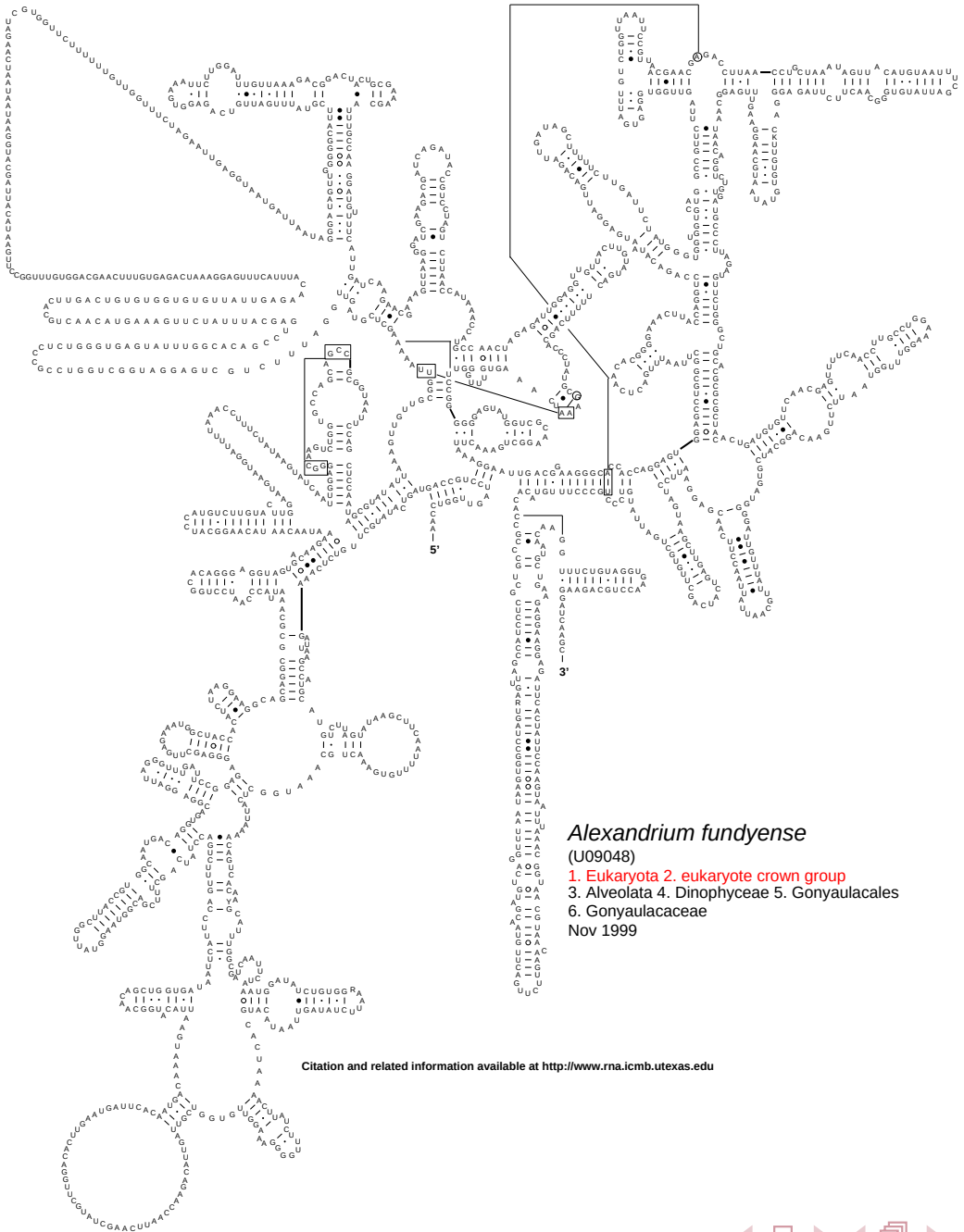
Ribsome

- In **Prokaryotes**, the ribosomes are designated as 70S
- 70S has two subunits: small (30S) and large (50S)
- **Smal subunit** consists of 16S (1540 nucleotides) + 21 proteins
- **Large subunit** consists of 5S (120 nucleotides), 23S (2900 nucleotides) + 34 proteins

5S



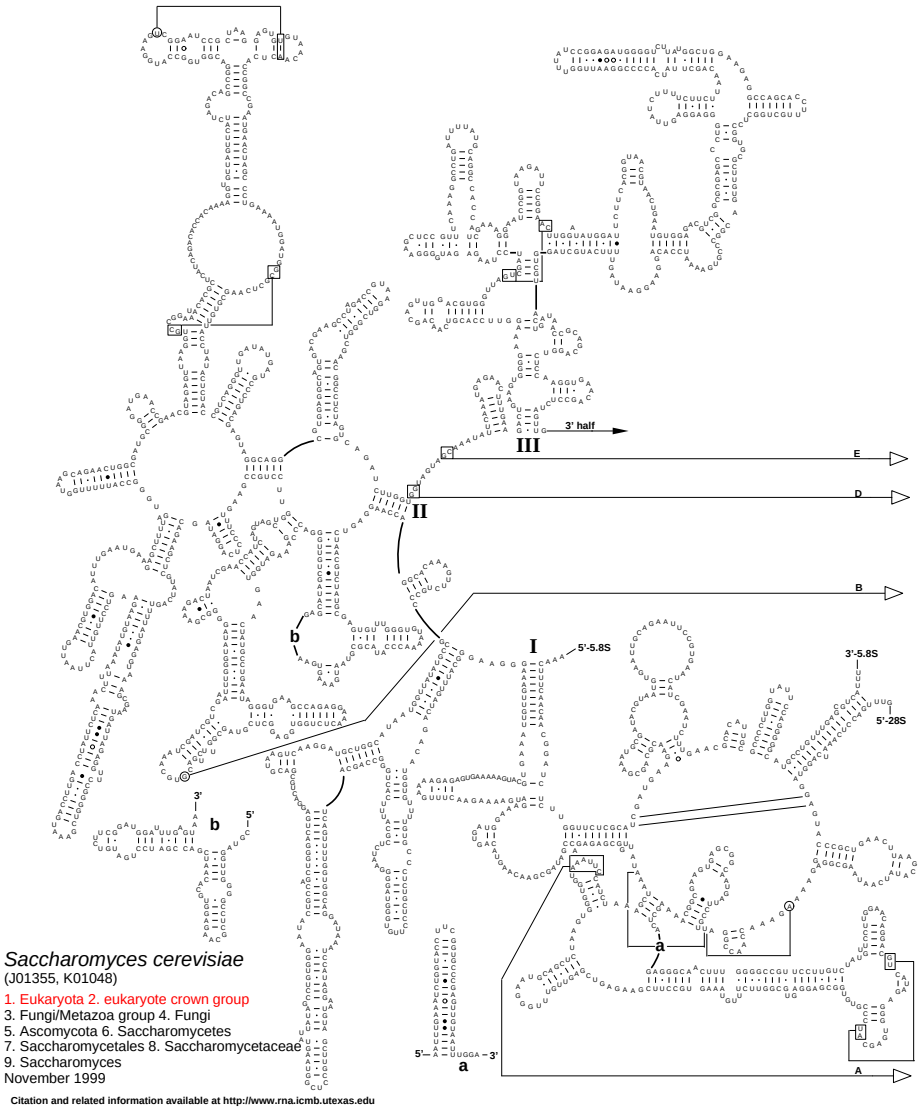
Secondary Structure: small subunit ribosomal RNA



Alexandrium fundyense
(U09048)
1. Eukaryota 2. eukaryote crown group
3. Alveolata 4. Dinophyceae 5. Gonyaulacales
6. Gonyaulacaceae
Nov 1999

Citation and related information available at http://www.rna.icmb.utexas.edu

Secondary Structure: large subunit ribosomal RNA - 5’ half



Saccharomyces cerevisiae
(J01355, K01048)

1. Eukaryota 2. eukaryote crown group
3. Fungi/Metazoa group 4. Fungi
5. Ascomycota 6. Saccharomycetes
7. Saccharomycetales 8. Saccharomycetaceae
9. Saccharomyces
November 1999

Citation and related information available at http://www.rna.icmb.utexas.edu

Secondary Structure: large subunit ribosomal RNA - 3’ half

Secondary Structure Prediction

- X ray **crystallography** and **N.M.R.**

Secondary Structure Prediction

- X ray **crystallography** and **N.M.R.**
- Chemical and enzymatic **probing**, **cross-linking**

Secondary Structure Prediction

- X ray **crystallography** and **N.M.R.**
- Chemical and enzymatic **probing**, **cross-linking**
- **Comparative sequence analysis**

Secondary Structure Prediction

- X ray **crystallography** and **N.M.R.**
- Chemical and enzymatic **probing**, **cross-linking**
- **Comparative sequence analysis**
- **Minimum free energy (MFE) methods**

Secondary Structure Prediction

- X ray **crystallography** and **N.M.R.**
- Chemical and enzymatic **probing**, **cross-linking**
- **Comparative sequence analysis**
- **Minimum free energy (MFE) methods**
- **Consensus (Comparative sequence analysis + MFE)**

human	AAGACUUCGGAUCUGGCGACACCC
mouse	ACACUUCGGAUGACACCAAAGUG
worm	AGGUCUUCGGCACGGGCACCAUUC
fly	CAACUUCGGAUUUUGCUACCAUA
orc	AAGCCUUCGGAGCGGGCGUAACU

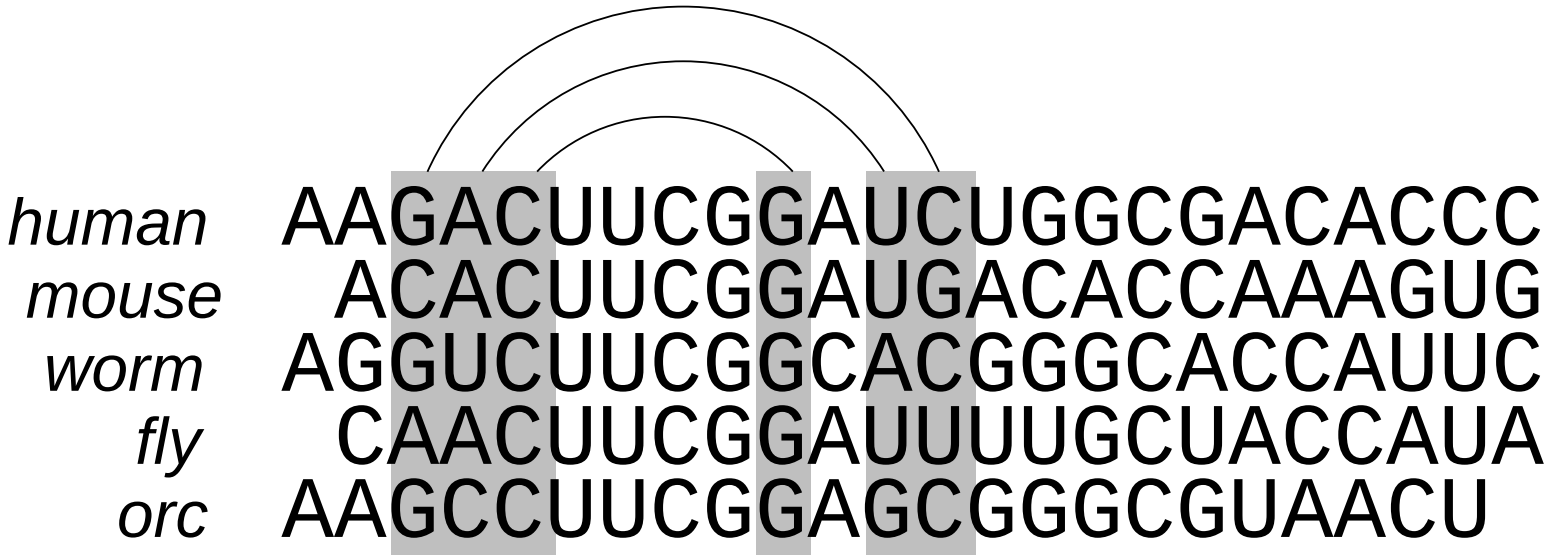
"Today, **comparative analysis** has become the method of choice for establishing higher-order structure for large RNA." Pace, Thomas, Woese (1999) In *The RNA World*. Cold Spring Harbor.

```
human   AAGACUUCGGAUCUGGCGACACCC
mouse     ACACUUCGGAUGACACCAAAGUG
worm    AGGUCUUCGGCACGGGCACCAUUC
fly       CAACUUCGGAUUUUGCUACCAUA
orc     AAGCCUUCGGAGCGGGCGUAACU
```

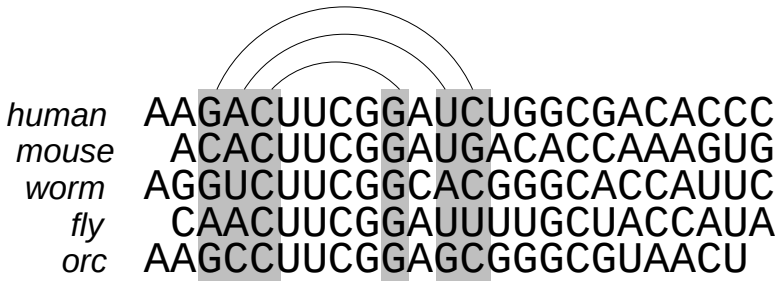
"Today, **comparative analysis** has become the method of choice for establishing higher-order structure for large RNA." Pace, Thomas, Woese (1999) In *The RNA World*. Cold Spring Harbor.

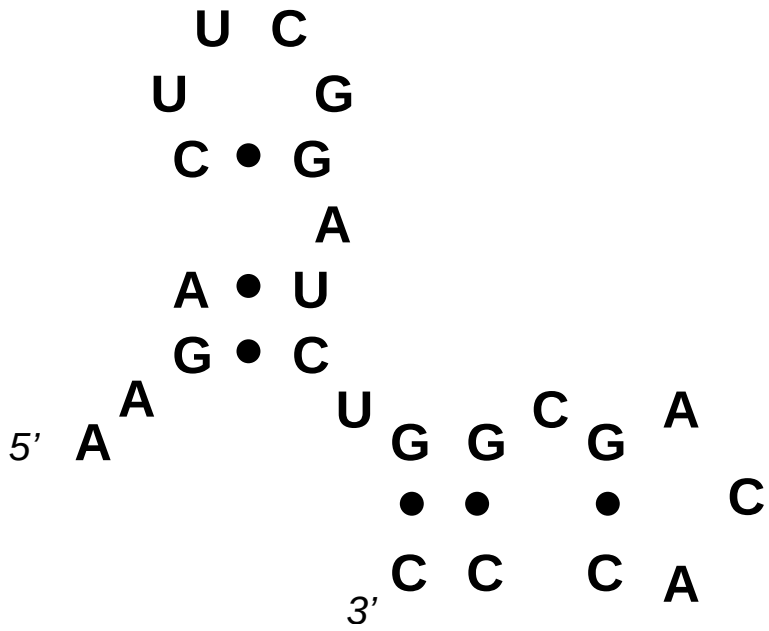
```
human   AAGACUUCGGAUCUGGCGACACCC
mouse     ACACUUCGGAUGACACCAAAGUG
worm    AGGUCUUCGGCACGGGCACCAUUC
fly       CAACUUCGGAUUUUGCUACCAUA
orc     AAGCCUUCGGAGCGGGCGUAACU
```

"Today, **comparative analysis** has become the method of choice for establishing higher-order structure for large RNA." Pace, Thomas, Woese (1999) In *The RNA World*. Cold Spring Harbor.



"Today, comparative analysis has become the method of choice for establishing higher-order structure for large RNA." Pace, Thomas, Woese (1999) In *The RNA World*. Cold Spring Harbor.


human AAGACUUCGGAUCUGGCGACACCC
mouse ACACUUCGGAUGACACCAAAGUG
worm AGGUCUUCGGCACGGGCACCAUUC
fly CAACUUCGGAUUUUGCUACCAUA
orc AAGCCUUCGGAGCGGGCGUAACU

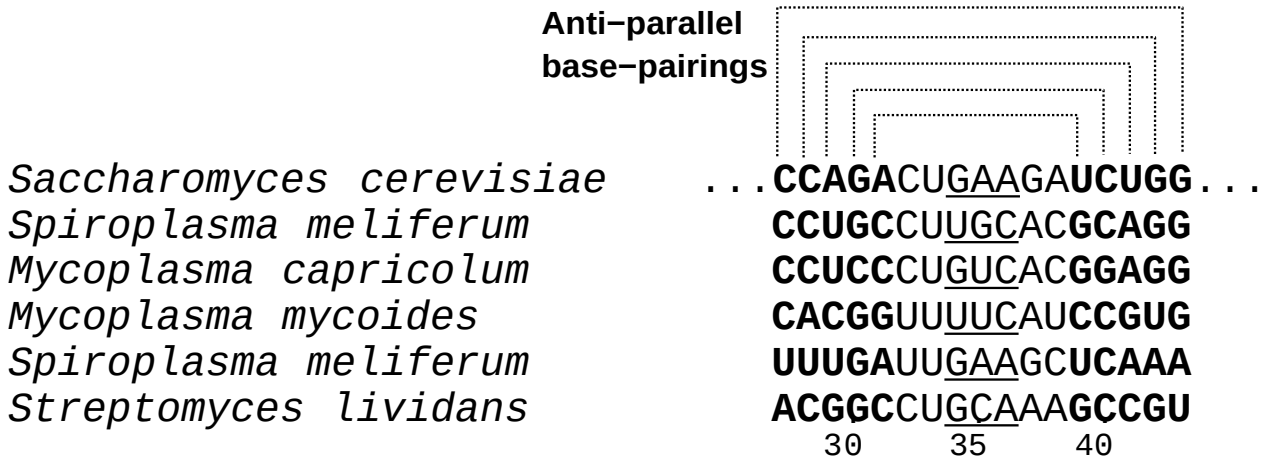

5'
3'

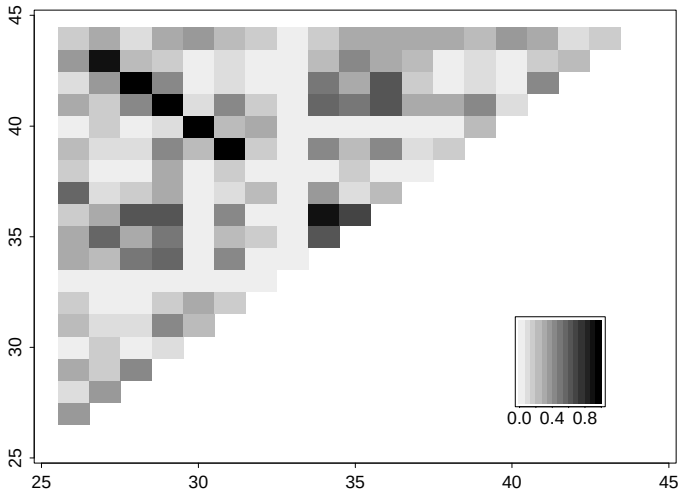
- Starts with the alignment of a set of homologous sequences

- Starts with the alignment of a set of homologous sequences
- Detecting correlated pairs of sites

- Starts with the alignment of a set of homologous sequences
- Detecting correlated pairs of sites
  - Parallel chords implies helices (stems)

- Starts with the alignment of a set of homologous sequences
- Detecting correlated pairs of sites
  - Parallel chords implies helices (stems)
  - Others are tertiary structure interactions

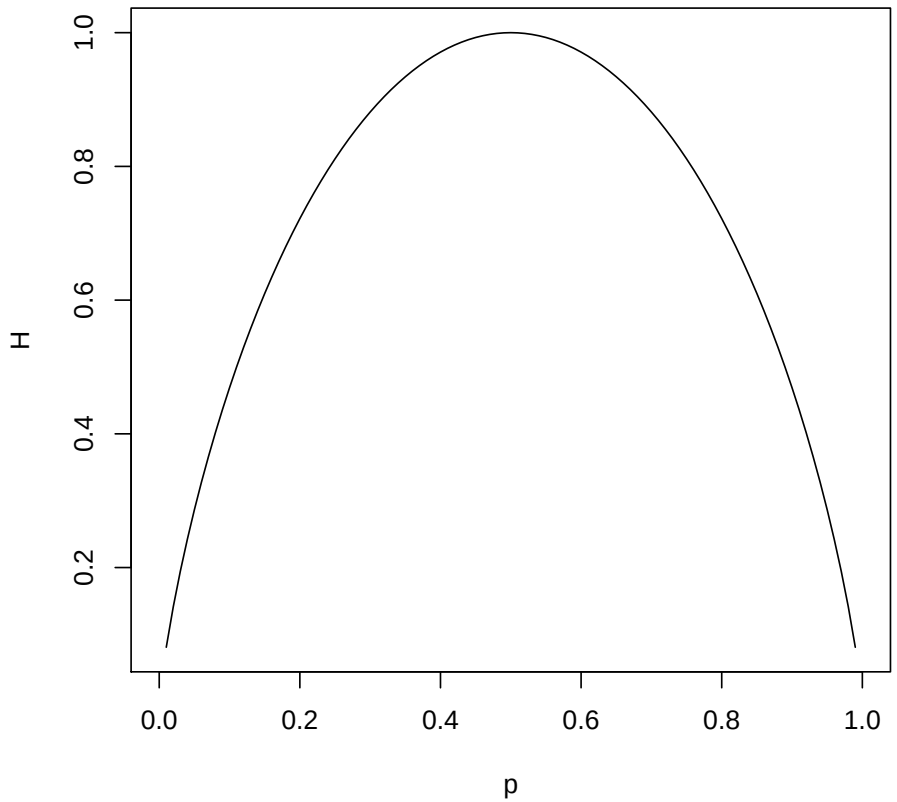

Anti−parallel
base−pairings
Saccharomyces cerevisiae ...CCAGACUGAAGAUCUGG...
Spiroplasma meliferum CCUGCCUUGCACGCAGG
Mycoplasma capricolum CCUCCCUGUCACGGAGG
Mycoplasma mycoides CACGGUUUUCAUCCGUG
Spiroplasma meliferum UUUGAUUGAAGCUCAAA
Streptomyces lividans ACGGCCUGCAAAGCCGU
30
35
40


45
40
35
30
25
25
30
35
40
45
0.0 0.4 0.8

Detecting **Correlated** Pairs

- **Matrix reduction**
  T Haselman, J E Chappelear and G E Fox (1988) Fidelity of secondary and tertiary interactions in tRNA. *Nucleic Acids Res.* **16**(12): 5673-5684.
- **Chi-square test of independence**
- **Measure of association** λ
  Goodman, Leo A. and Kruskal, William H. (1979) Measures of association for cross classifications. New York, Springer-Verlag.
- **Mutual information**
  - $M(I,J) = H(I) + H(J) - H(I,J)$
    where $H(I) = -\sum_\alpha P(i=\alpha) \log P(i=\alpha)$
    and $H(I,J) = -\sum_{\alpha\beta} P(i=\alpha, j=\beta) \log P(i=\alpha, j=\beta)$

Entropy or **uncertainty**, one variable, two outcomes



The above picture shows how the **entropy** varies as a function of p.

Accuracy of comparative analysis on rRNAs

- Late **1970**'s, comparative sequence analysis
- 16S $\sim$ 1500 nt long, 23S $\sim$ 3000 nt long
- 4.3×10^{393} and 6.3×10^{740} possible secondary structures
- **2000**, high-resolution crystal structures of rRNAs produced
- Gutell et al. The accuracy of ribosomal RNA comparative structure models. Curr Opin Struct Biol (2002) vol. 12 (3) pp. 301-10
- "**97–98% of the base pairings predicted with covariation analysis are indeed present in the 16S and 23S rRNA crystal structures**"

What are the main **difficulties**?

What are the main **difficulties**?

- Needs an alignment, but sequence alignment techniques are not well adapted for RNA sequences

What are the main **difficulties**?

- Needs an alignment, but sequence alignment techniques are not well adapted for RNA sequences
- To produce a high quality alignment, the sequences should be similar

What are the main **difficulties**?

- Needs an alignment, but sequence alignment techniques are not well adapted for RNA sequences
- To produce a high quality alignment, the sequences should be similar
- If the sequences are similar, there will be few observed compensatory changes

RNA Secondary Structure **Prediction**

- Considering a sequence 200 nucleotides long there are on the order of 10^{50} possible secondary structures! Durbin *et et al.* page 267

RNA Secondary Structure **Prediction**

- Considering a sequence 200 nucleotides long there are on the order of 10^{50} possible secondary structures! Durbin *et et al.* page 267
- How to search the space?

RNA Secondary Structure **Prediction**

- Considering a sequence 200 nucleotides long there are on the order of 10^{50} possible secondary structures! Durbin *et et al.* page 267
- How to search the space?
  - Nussinov: a didactic example

RNA Secondary Structure **Prediction**

- Considering a sequence 200 nucleotides long there are on the order of 10^{50} possible secondary structures! Durbin *et et al.* page 267
- How to search the space?
  - Nussinov: a didactic example
- RNAs adopt one (or a few) stable structure(s). Which one?

RNA Secondary Structure **Prediction**

- Considering a sequence 200 nucleotides long there are on the order of 10^{50} possible secondary structures! Durbin *et et al.* page 267
- How to search the space?
  - Nussinov: a didactic example
- RNAs adopt one (or a few) stable structure(s). Which one?
  - Zuker: minimizing the total free energy

RNA Secondary Structure **Determination**

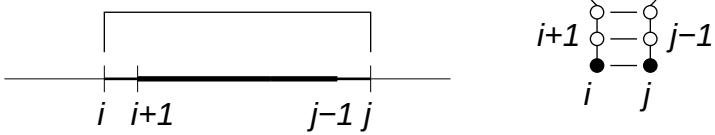
A didactic example first. Nussinov's algorithm finds **the structure that maximises the total number of base pairs**.

RNA Secondary Structure **Determination**

A didactic example first. Nussinov's algorithm finds **the structure that maximises the total number of base pairs**.
A well behaved problem!

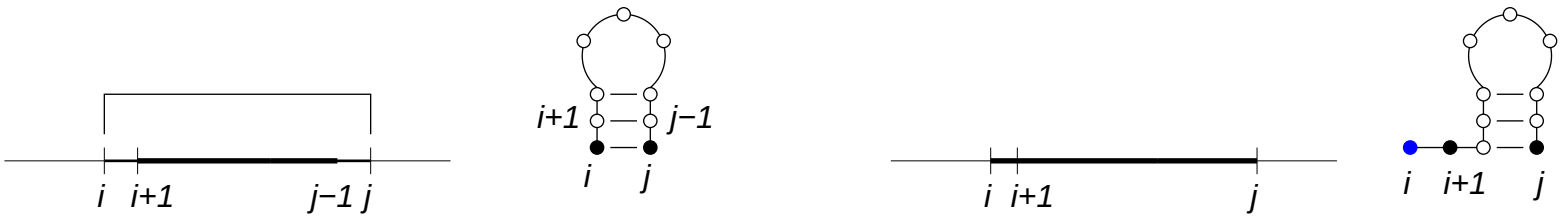
RNA Secondary Structure **Determination**

A didactic example first. Nussinov's algorithm finds **the structure that maximises the total number of base pairs**.
A well behaved problem!



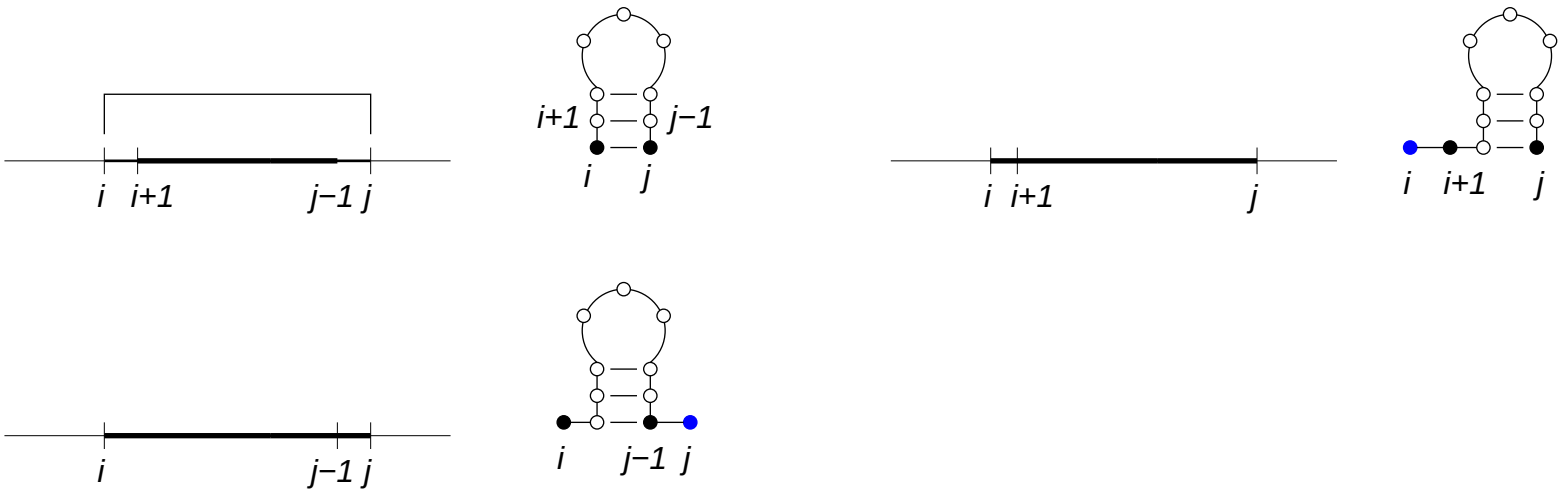
RNA Secondary Structure **Determination**

A didactic example first. Nussinov's algorithm finds **the structure that maximises the total number of base pairs**.
A well behaved problem!



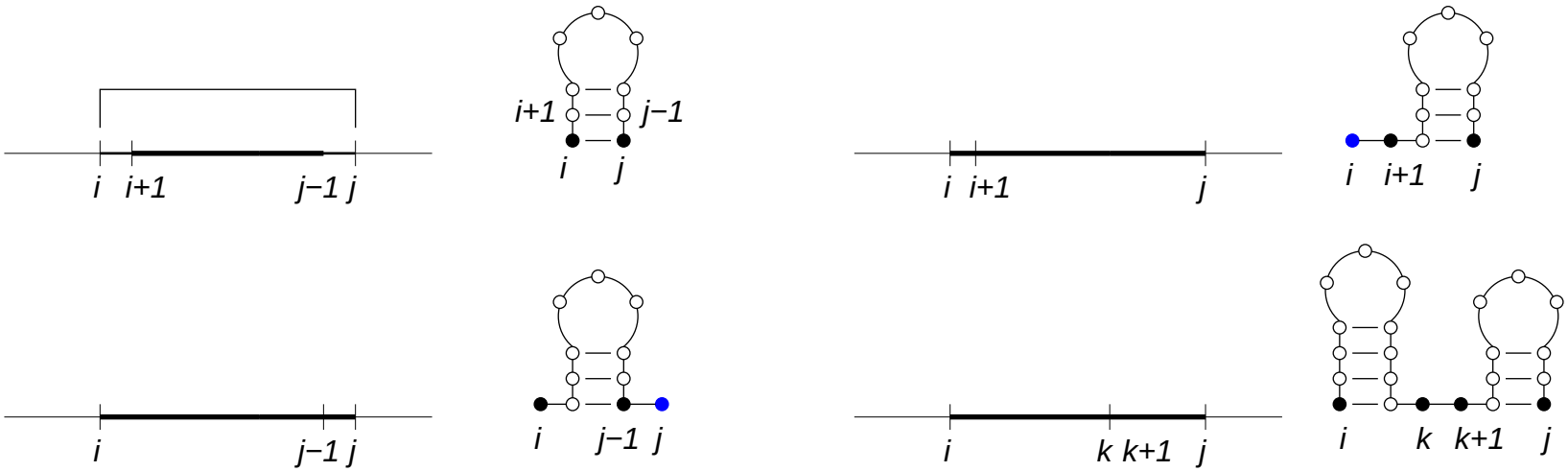
RNA Secondary Structure **Determination**

A didactic example first. Nussinov's algorithm finds **the structure that maximises the total number of base pairs**.
A well behaved problem!



RNA Secondary Structure **Determination**

A didactic example first. Nussinov's algorithm finds **the structure that maximises the total number of base pairs**.
A well behaved problem!



Nussinov Algorithm

Initialisation:

$$\gamma(i, i+k) \quad = \quad 0 \quad \text{for k} = 0 \text{ to } 2 \text{ and for i} = 1 \text{ to n} - \text{k}.$$

Nussinov Algorithm

Initialisation:

$$\gamma(i, i+k) \quad = \quad 0 \quad \text{for k} = 0 \text{ to } 2 \text{ and for i} = 1 \text{ to n} - \text{k}.$$

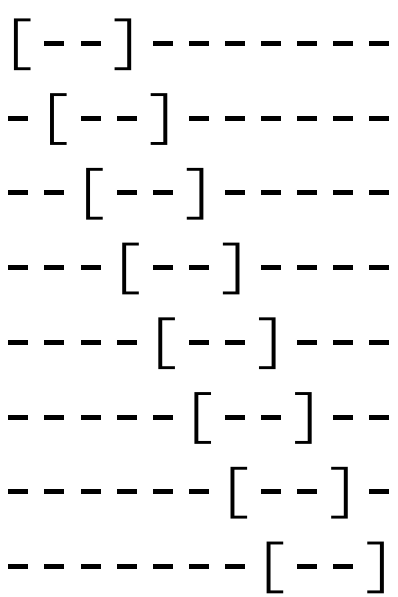
Recurrence:

$$\gamma(i,j) = \max \begin{cases} \gamma(i+1, j-1) + \delta(i,j); \\ \gamma(i+1, j); \\ \gamma(i, j-1); \\ \max_{i<k<(j-1)} [\gamma(i,k) + \gamma(k+1, j)]. \end{cases}$$

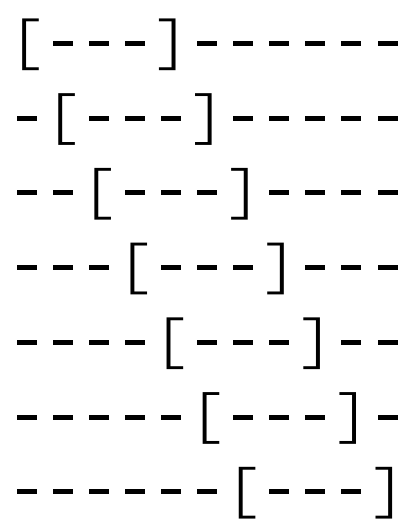
Matching score:

$$\delta(i,j) = \begin{cases} 1, \text{if } a_i : a_j \in \{A:U, U:A, G:C, C:G\} \bigcup \{G:U, U:G\}; \\ 0, \text{otherwise}. \end{cases}$$

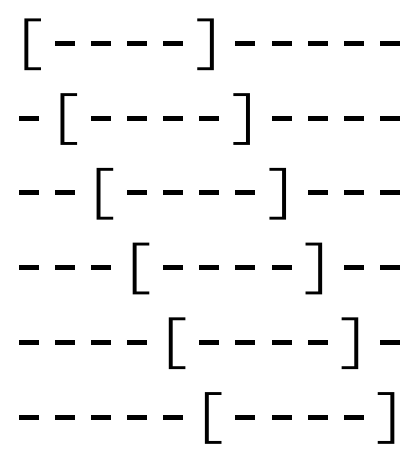
$j-i+1=2$



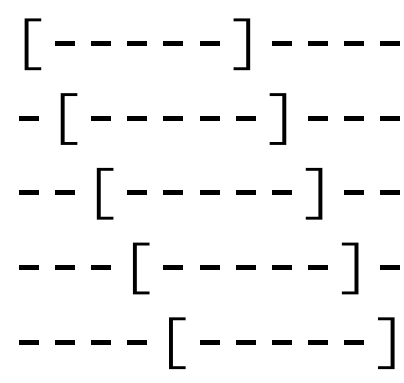
$j-i+1=3$



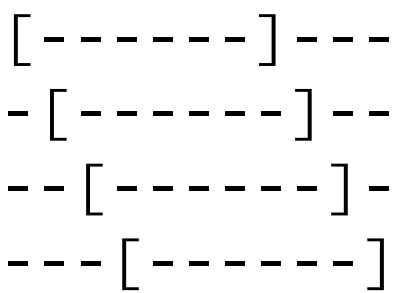
$j-i+1=4$



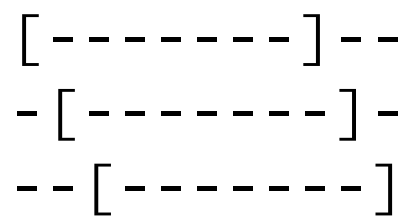
$j-i+1=5$



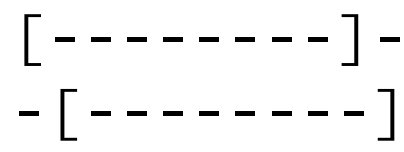
$j-i+1=6$



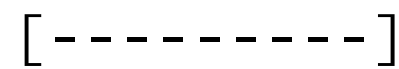
$j-i+1=7$



$j-i+1=8$



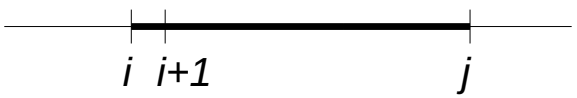
$j-i+1=9$

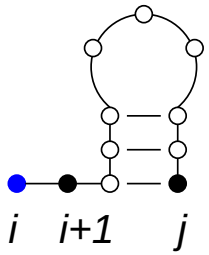


	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

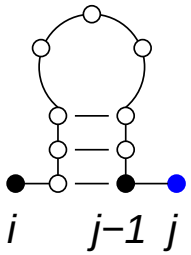
$\Rightarrow$ Initialization (blue values)

	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

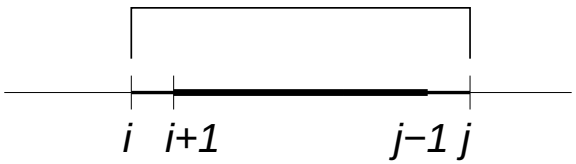


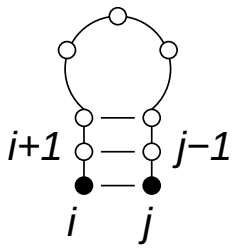


	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

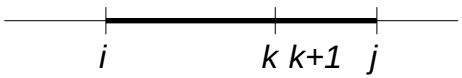


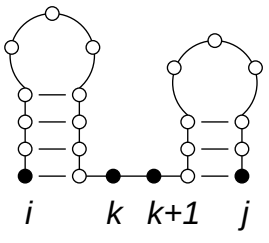
	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0





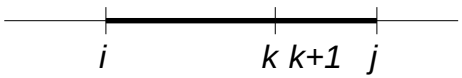
	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

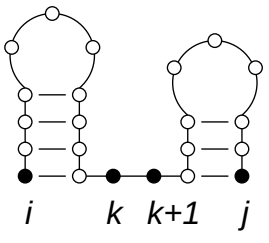





```
[-][-------]
i k        j
```

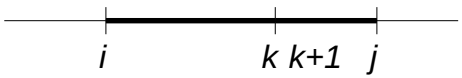
	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

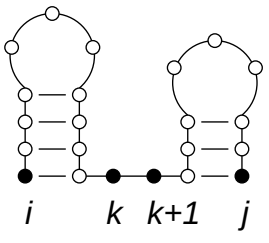





```
[--][------]
i   k      j
```

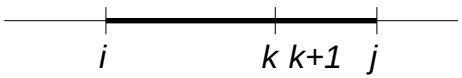
	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

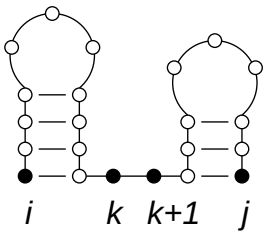





```
[---][-----]
i   k      j
```

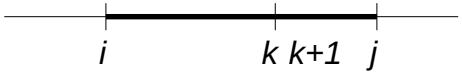
	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

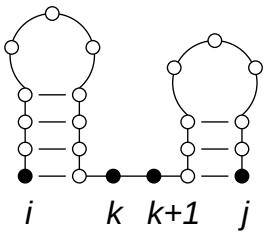





```
[----][----]
i    k    j
```

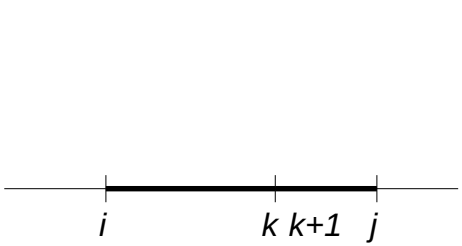
	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

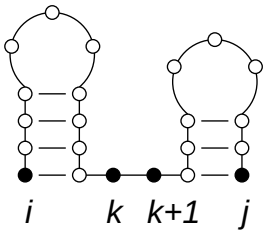





```
[-----][---]
i     k    j
```

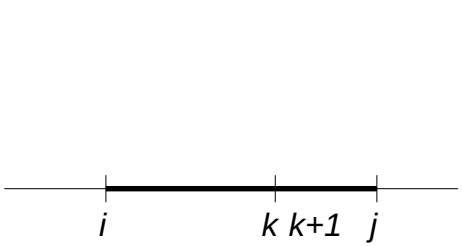
	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

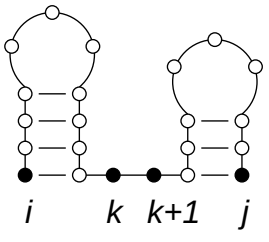





```
[------][--]
i      k   j
```

	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0






```
[-------][-]
i       k  j
```

	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	1	2	3
G	0	0	0	0	0	0	1	2	3
G		0	0	0	0	0	1	2	2
A			0	0	0	0	1	1	1
A				0	0	0	1	1	1
A					0	0	1	1	1
U						0	0	0	0
C							0	0	0
C								0	0

Traceback

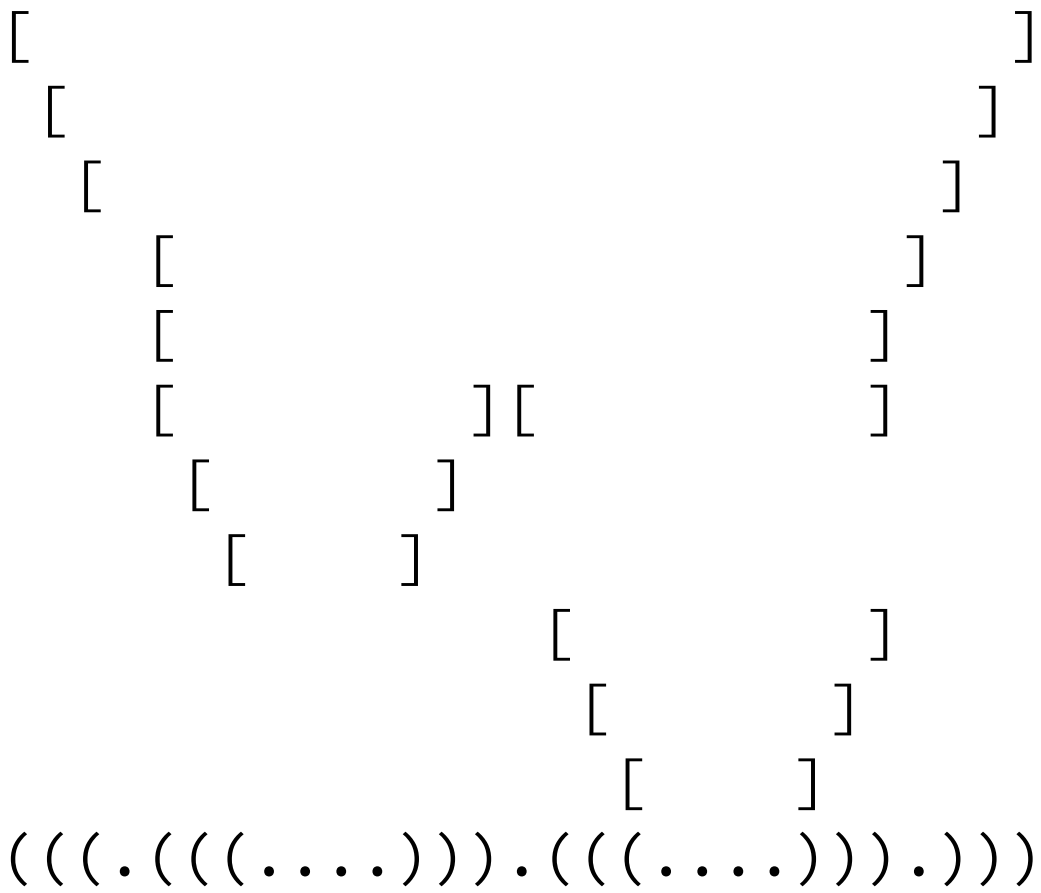
How?

Traceback

How?

```
private String traceback(int i, int j) {
   if (g[i][j] == 0)
        return stringRepeat('.', j-i+1);
   if (g[i][j] == g[i+1][j-1] + delta(a.charAt(i),a.char
        return "(" + traceback(i+1,j-1) + ")";
   if (g[i][j] == g[i+1][j])
        return "." + traceback(i+1,j);
   if (g[i][j] == g[i][j-1])
        return traceback(i,j-1) + ".";
   for (int k=i+1; k<j; k++)
        if (g[i][j] == g[i][k]+g[k+1][j])
            return traceback(i,k) + traceback(k+1,j);
}
```

Traceback algorithm



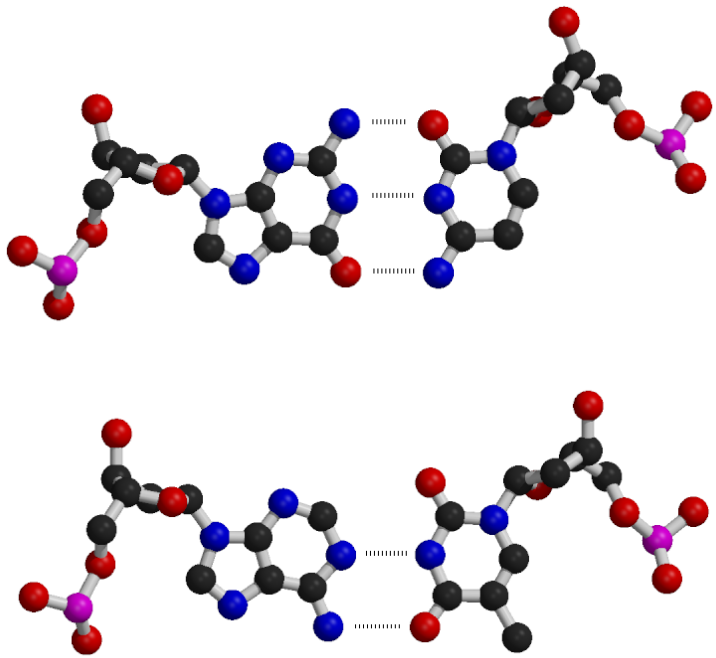
Traceback algorithm

```
{ Initialization }
  push(1, N)

{ Main loop }
  while pop(i, j)
      if  γ(i+1,j) = γ(i,j)
          push(i+1,j)
      else if  γ(i,j−1) = γ(i,j)
          push(i,j−1)
      else if  γ(i+1, j+1) + δ(i,j) = γ(i,j)
          write base-pair i,j
          push  (i+1, j−1)
      else for k = i+1 to j−1
          if γ(i,k) + γ(k+1,j) = γ(i,j)
              push(k+1, j)
              push(i,k)
```

Remarks

- What is the time and space complexity?
- **Maximum number of base pairs** is not a good objective function!

Is maximizing the number of hydrogen bonds a better objective function?



Maximising the number of hydrogen bonds

Initialisation:

$$\gamma(i, i+k) \quad = \quad 0 \quad \text{for k} = 0 \text{ to } 2 \text{ and for i} = 1 \text{ to n} - \text{k}.$$

Recurrence:

$$\gamma(i,j) = \max \begin{cases} \gamma(i+1, j-1) + \delta(i,j); \\ \gamma(i+1, j); \\ \gamma(i, j-1); \\ \max_{i<k<(j-1)} [\gamma(i,k) + \gamma(k+1, j)]. \end{cases}$$

Matching score:

$$\delta(i,j) = \begin{cases} 3, \text{if } a_i : a_j \in \{G:C, C:G\}; \\ 2, \text{if } a_i : a_j \in \{A:U, U:A\}; \\ 1, \text{if } a_i : a_j \in \{G:U, U:G\}; \\ 0, \text{otherwise.} \end{cases}$$

Nussinov: Summary

- **Maximum number of hydrogen bonds** is also a bad objective function

Nussinov: Summary

- **Maximum number of hydrogen bonds** is also a bad objective function
- Space complexity $O(L^2)$

Nussinov: Summary

- **Maximum number of hydrogen bonds** is also a bad objective function
- Space complexity $O(L^2)$
- Time complexity $O(L^3)$

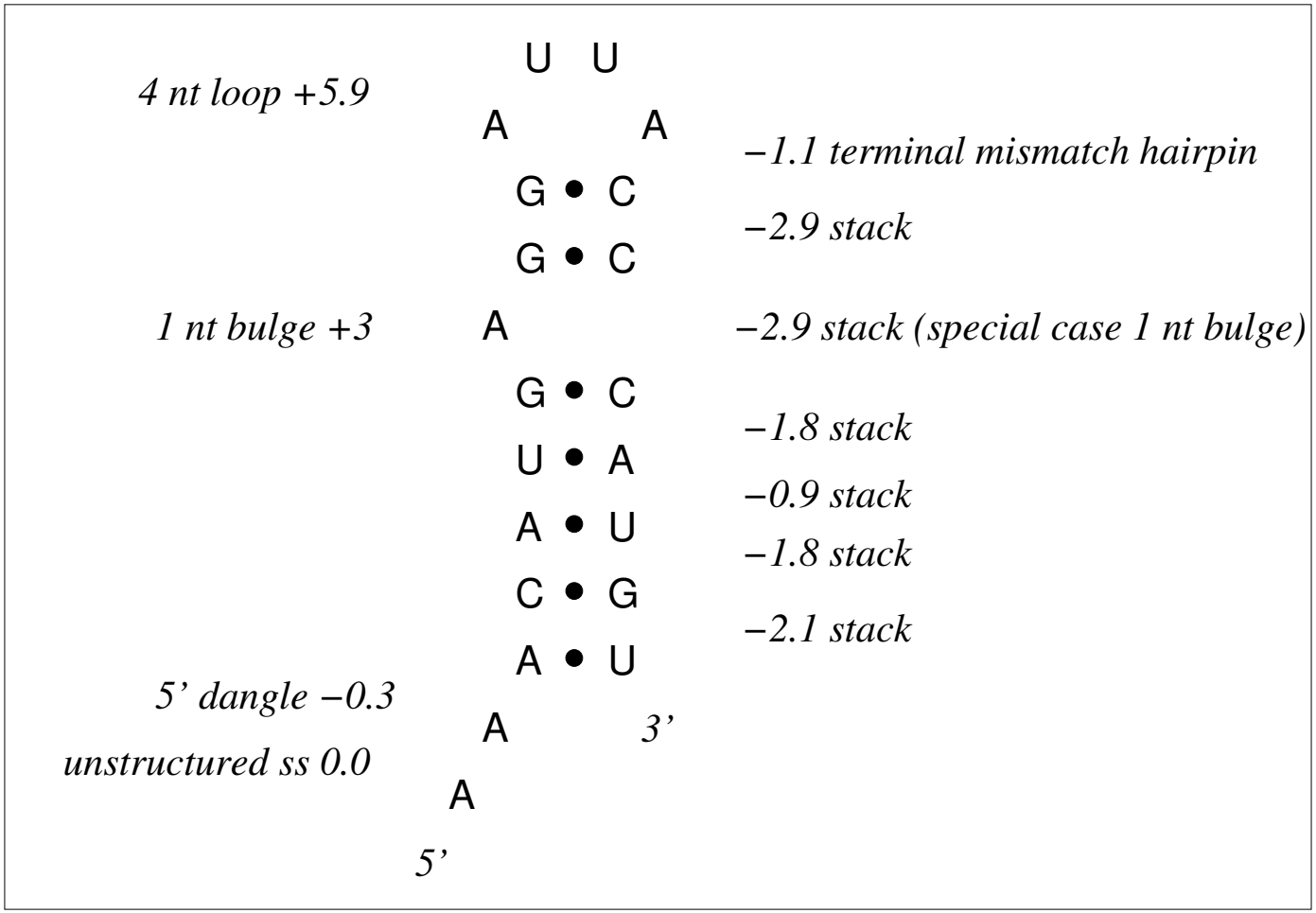
Nussinov: Summary

- **Maximum number of hydrogen bonds** is also a bad objective function
- Space complexity $O(L^2)$
- Time complexity $O(L^3)$
- Does not model “real” structures well enough

Nussinov: Summary

- **Maximum number of hydrogen bonds** is also a bad objective function
- Space complexity $O(L^2)$
- Time complexity $O(L^3)$
- Does not model "real" structures well enough
- But works similarly to the Zuker algorithm

Nearest-neighbor model



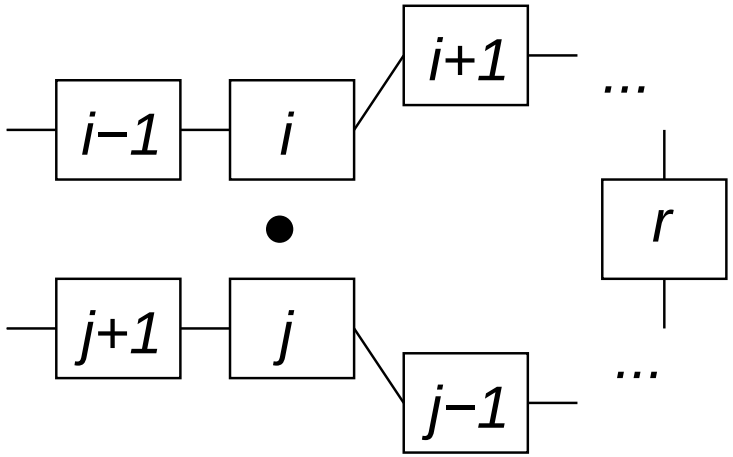
Free **Engergy**

In thermodynamics, the term free energy denotes either of two related concepts of importance. They express the total amount of energy which is used up or released during a chemical reaction. Both attempt to capture that part of the total energy of a system which is available for "useful work" and is hence not stored in "useless random thermal motion". As a system undergoes changes, its free energy will decrease.

Wikipedia

Obervations and notation: **hairpin loop**

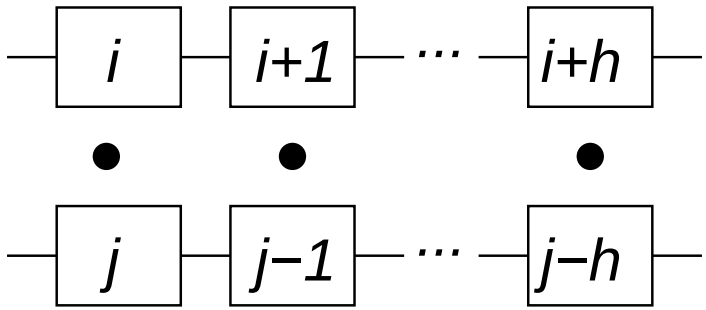
If $i \bullet j$ is a pair and $i < r < j$, we say that $i \bullet j$ *surrounds* r.



If S contains $i \bullet j$ but none of the surrounded nucleotides $i + 1$ to $j - 1$ are paired the result is a **hairpin loop**.

Obervations and notation: **stacked pair**

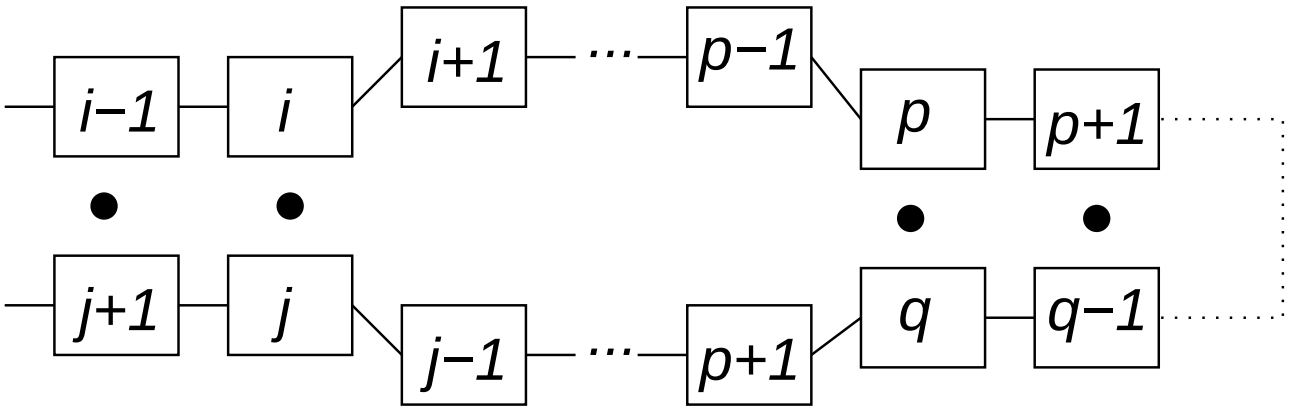
Given 2 pairs, $i \bullet j$ and $p \bullet q$, if $i \bullet j$ surrounds either p or q then, because no knots are allowed, $i \bullet j$ surrounds both p and q.



If S contains $i \bullet j$, $(i+1) \bullet (j-1), \ldots, (i+h) \bullet (j-h)$, each of these pairs (except the last one) is said to stack onto the following pair. Two consecutive pairs are referred to as **stacked pair** or **stacked pair cycle**.

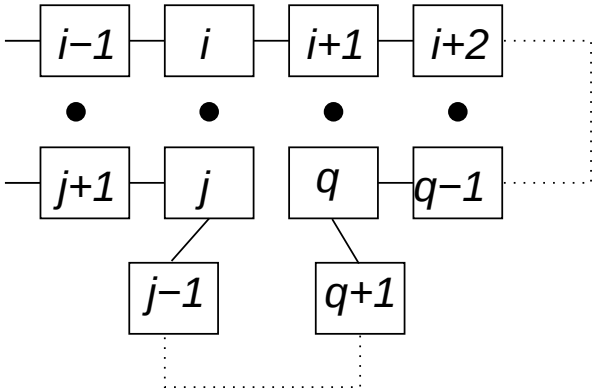
Obervations and notation: **interior loop**

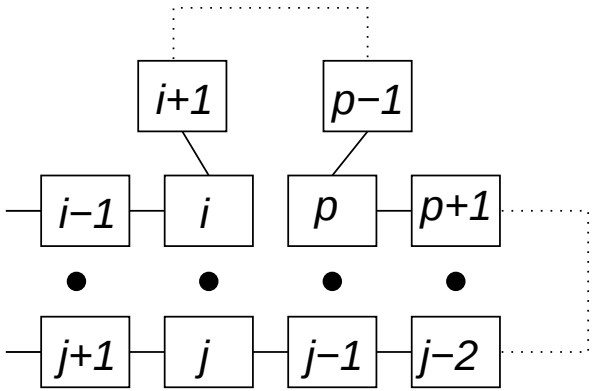
Given a pair $i \bullet j$ surrounding another pair $p \bullet q$.



If the elements between i and p are unpaired and the elements between q and j are also unpaired the resulting structure is called an **interior loop**.

Obervations and notation: **bulge**



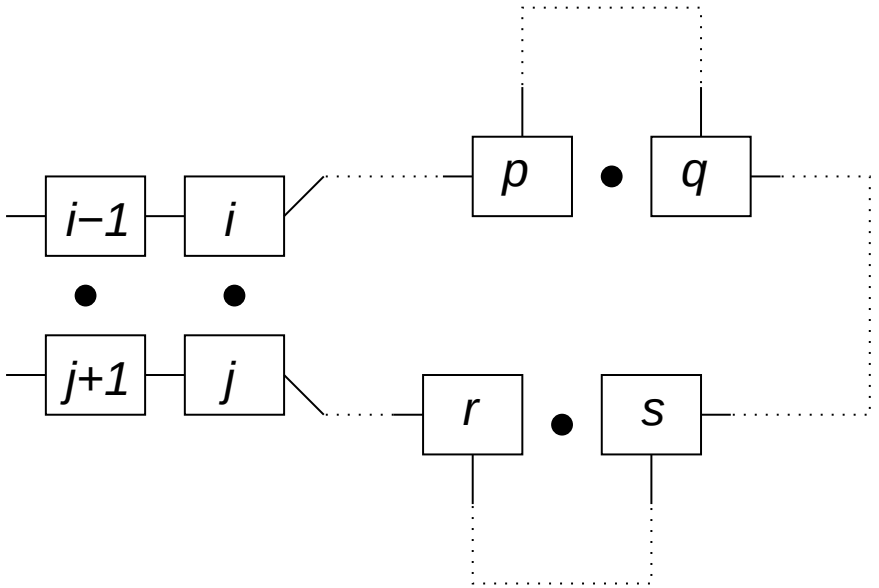


Given 2 pairs, $i \bullet j$ and $(i+1) \bullet q$. If there are some unpaired elements between q and j then these elements are said to form a **bulge**.

Symmetrically, the unpaired elements $i+1 \ldots p-1$ can form a **bulge** if S contains $i \bullet j$ and $p \bullet (j-1)$.

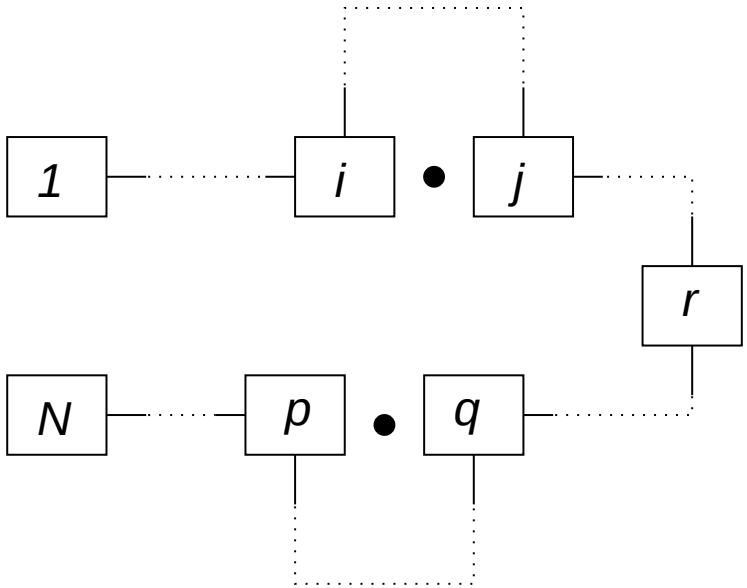
Obervations and notation: **multiple loop**

Given S containing the pair $i \bullet j$.



If $i \bullet j$ surrounds 2 or more pairs, $p \bullet q$ and $r \bullet s$, which do not surround one another, the resulting structure is a **multiple loop**.

Obervations and notation: **single stranded** region



If S contains r but there are not surrounding pair, we say that r belongs to a **single stranded region** or external region.

Observations and notation: **accessible** in/from

- A pair $p \bullet q$ or an unpaired element r is **accessible in** $[i, j]$ if it is not surrounded by any pair except possibly $i \bullet j$.
- If i and j are paired $p \bullet q$ or r are said to be **accessible from** $i \bullet j$.

Observations and notation: **cycle**, **order**

- Any pair $i \bullet j$ defines a **cycle s** from i to j: **s** consists of the **closing pair** $i \bullet j$, together with any pairs $p_1 \bullet q_1, p_2 \bullet q_2$ **accessible from** $i \bullet j$ and any unpaired elements **accessible from** $i \bullet j$.

Observations and notation: **cycle**, **order**

- Any pair $i \bullet j$ defines a **cycle s** from i to j: **s** consists of the **closing pair** $i \bullet j$, together with any pairs $p_1 \bullet q_1, p_2 \bullet q_2$ **accessible from** $i \bullet j$ and any unpaired elements **accessible from** $i \bullet j$.
- If **s** contains k pairs, including the closing pair, then **s** is said to be a *k-cycle* (or to have order k).

Observations and notation: **cycle**, **order**

- Any pair $i \bullet j$ defines a **cycle s** from i to j: **s** consists of the **closing pair** $i \bullet j$, together with any pairs $p_1 \bullet q_1, p_2 \bullet q_2$ **accessible from** $i \bullet j$ and any unpaired elements **accessible from** $i \bullet j$.
- If **s** contains k pairs, including the closing pair, then **s** is said to be a *k-cycle* (or to have order k).
- A secondary structure on $[i, j]$ is indicated by S_{ij}.

Relationships between k-**cycles** and **structures**

Given a cycle of order k from i to j and p and q such that $(i+1) < p$ and $p < q$ and $q < (j-1)$, it follows that:

Relationships between k-**cycles** and **structures**

Given a cycle of order k from i to j and p and q such that $(i+1) < p$ and $p < q$ and $q < (j-1)$, it follows that:

- If $k = 1$, the 1-cycle is a **hairpin**

Relationships between k-**cycles** and **structures**

Given a cycle of order k from i to j and p and q such that $(i+1) < p$ and $p < q$ and $q < (j-1)$, it follows that:

- If $k = 1$, the 1-cycle is a **hairpin**
- If $k = 2$ and the accessible pair is $(i+1) \bullet (j-1)$, then this 2-cycle is a **stacked pair** (cycle)

Relationships between k-**cycles** and **structures**

Given a cycle of order k from i to j and p and q such that $(i+1) < p$ and $p < q$ and $q < (j-1)$, it follows that:

- If $k = 1$, the 1-cycle is a **hairpin**
- If $k = 2$ and the accessible pair is $(i+1) \bullet (j-1)$, then this 2-cycle is a **stacked pair** (cycle)
- If $k = 2$ and the accessible pair is $p \bullet q$, then the 2-cycle is an **interior loop**

Relationships between k-**cycles** and **structures**

Given a cycle of order k from i to j and p and q such that $(i+1) < p$ and $p < q$ and $q < (j-1)$, it follows that:

- If $k = 1$, the 1-cycle is a **hairpin**
- If $k = 2$ and the accessible pair is $(i+1) \bullet (j-1)$, then this 2-cycle is a **stacked pair** (cycle)
- If $k = 2$ and the accessible pair is $p \bullet q$, then the 2-cycle is an **interior loop**
- If $k = 2$ and the accessible pair is either $(i+1) \bullet q$ or $p \bullet (j-1)$, then the 2-cycle is a **bulge**

Relationships between k-**cycles** and **structures**

Given a cycle of order k from i to j and p and q such that $(i+1) < p$ and $p < q$ and $q < (j-1)$, it follows that:

- If $k = 1$, the 1-cycle is a **hairpin**
- If $k = 2$ and the accessible pair is $(i+1) \bullet (j-1)$, then this 2-cycle is a **stacked pair** (cycle)
- If $k = 2$ and the accessible pair is $p \bullet q$, then the 2-cycle is an **interior loop**
- If $k = 2$ and the accessible pair is either $(i+1) \bullet q$ or $p \bullet (j-1)$, then the 2-cycle is a **bulge**
- If $k \geq 3$ then the k-cycle is a **multiple loop**

Tinoco-Uhlenbeck Theory

- The free energy can be expressed as sum of terms where each term represents a cycle, s_r:

$$E(S) = e(s_1) + e(s_2) + \ldots + e(s_t).$$

Tinoco-Uhlenbeck Theory

- The free energy can be expressed as sum of terms where each term represents a cycle, s_r:

$$E(S) = e(s_1) + e(s_2) + \ldots + e(s_t).$$

- Furthermore, $e(s) < 0$ when s is a stacked pair, hairpins and other loops make positive contributions

Tinoco-Uhlenbeck Theory

- The free energy can be expressed as sum of terms where each term represents a cycle, s_r:

$$E(S) = e(s_1) + e(s_2) + \ldots + e(s_t).$$

- Furthermore, $e(s) < 0$ when s is a stacked pair, hairpins and other loops make positive contributions
- An RNA sequence will tend to fold into a conformation that minimizes the free energy

Tinoco-Uhlenbeck Theory

- The free energy can be expressed as sum of terms where each term represents a cycle, s_r:

$$E(S) = e(s_1) + e(s_2) + \ldots + e(s_t).$$

- Furthermore, $e(s) < 0$ when s is a stacked pair, hairpins and other loops make positive contributions
- An RNA sequence will tend to fold into a conformation that minimizes the free energy
- Helical regions stabilize the molecule

Tinoco-Uhlenbeck Theory

- The free energy can be expressed as sum of terms where each term represents a cycle, s_r:

$$E(S) = e(s_1) + e(s_2) + \ldots + e(s_t).$$

- Furthermore, $e(s) < 0$ when s is a stacked pair, hairpins and other loops make positive contributions
- An RNA sequence will tend to fold into a conformation that minimizes the free energy
- Helical regions stabilize the molecule
- Loop regions have a de-stabilizing effect

Tinoco-Uhlenbeck Theory

- The free energy can be expressed as sum of terms where each term represents a cycle, s_r:

$$E(S) = e(s_1) + e(s_2) + \ldots + e(s_t).$$

- Furthermore, $e(s) < 0$ when s is a stacked pair, hairpins and other loops make positive contributions
- An RNA sequence will tend to fold into a conformation that minimizes the free energy
- Helical regions stabilize the molecule
- Loop regions have a de-stabilizing effect
- Single stranded (external) regions make no contribution, $e(s) = 0$

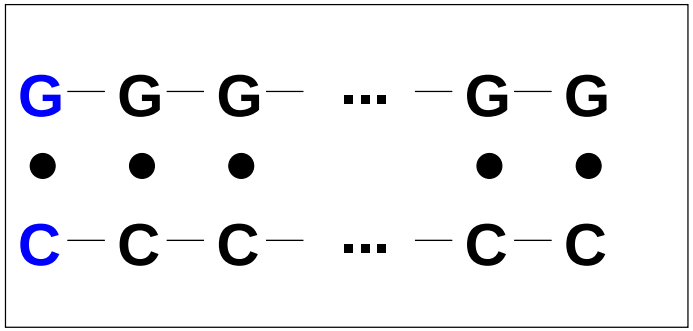
Tinoco-Uhlenbeck Theory

- The free energy can be expressed as sum of terms where each term represents a cycle, s_r:

$$E(S) = e(s_1) + e(s_2) + \ldots + e(s_t).$$

- Furthermore, $e(s) < 0$ when s is a stacked pair, hairpins and other loops make positive contributions
- An RNA sequence will tend to fold into a conformation that minimizes the free energy
- Helical regions stabilize the molecule
- Loop regions have a de-stabilizing effect
- Single stranded (external) regions make no contribution, $e(s) = 0$

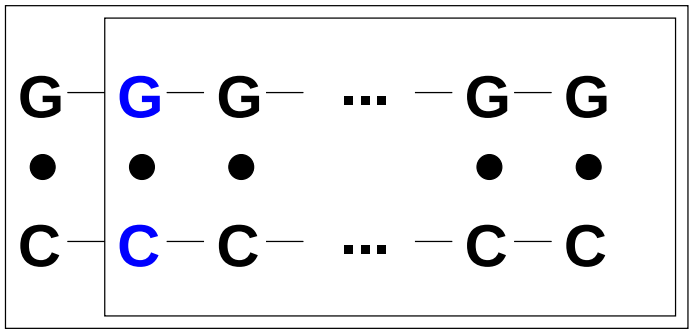
Tinoco-Uhlenbeck Theory



$E(s) = e(s_1) + \ldots$

$\Rightarrow$ Tinoco I. *et al* (1973) Improved estimation of secondary structure in ribonucleic acids. *Nature New Biology* **246**:40–41.

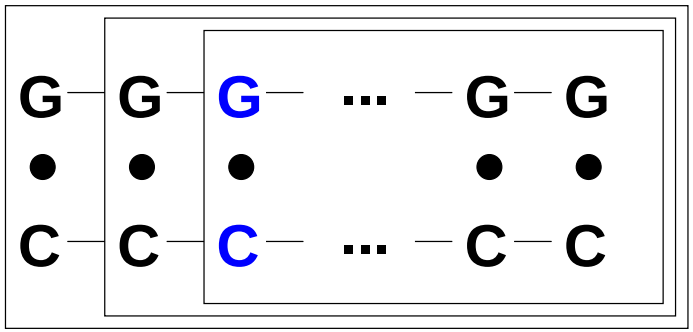
Tinoco-Uhlenbeck Theory



$$E(s) = e(s_1) + e(s_2) + \dots$$

$\Rightarrow$ Tinoco I. *et al* (1973) Improved estimation of secondary structure in ribonucleic acids. *Nature New Biology* **246**:40–41.

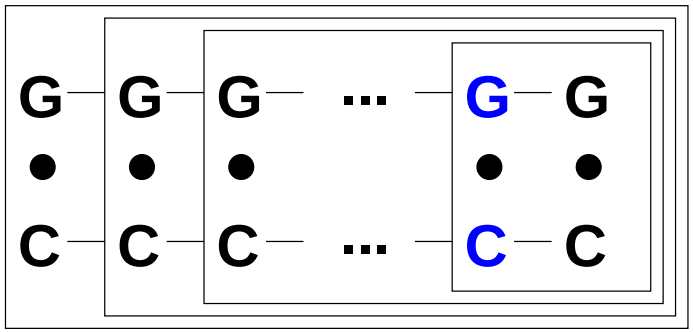
Tinoco-Uhlenbeck Theory



$$E(s) = e(s_1) + e(s_2) + e(s_3) + \dots$$

$\Rightarrow$ Tinoco I. *et al* (1973) Improved estimation of secondary structure in ribonucleic acids. *Nature New Biology* **246**:40–41.

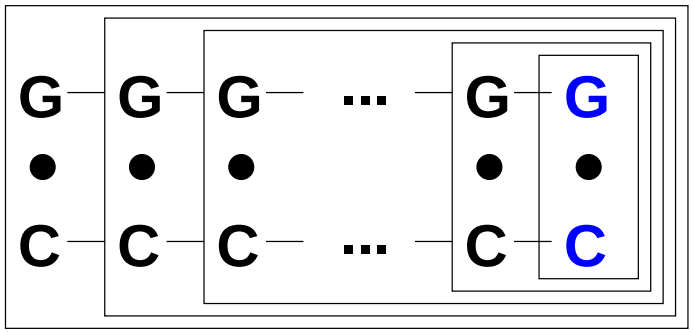
Tinoco-Uhlenbeck Theory



$$E(s) = e(s_1) + e(s_2) + e(s_3) + e(s_{t-1}) + \dots$$

$\Rightarrow$ Tinoco I. *et al* (1973) Improved estimation of secondary structure in ribonucleic acids. *Nature New Biology* **246**:40–41.

Tinoco-Uhlenbeck Theory



$$E(s) = e(s_1) + e(s_2) + e(s_3) + e(s_{n-1}) + e(s_t)$$

$\Rightarrow$ Tinoco I. *et al* (1973) Improved estimation of secondary structure in ribonucleic acids. *Nature New Biology* **246**:40–41.

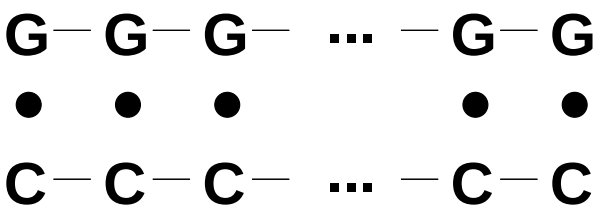
Estimating $e(s)$ terms

G–G–G– … –G–G

C–C–C– … –C–C

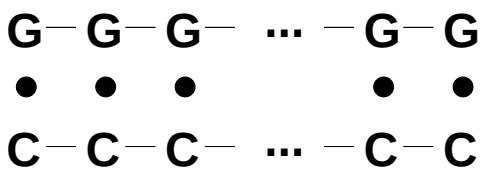
A chemical solution is prepared containing two complementary chains: GGG …G and CCC …C.

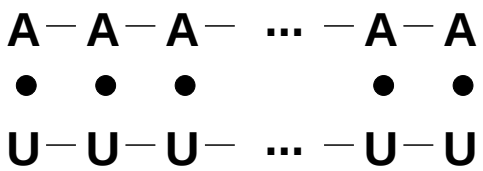
Estimating $e(s)$ terms



Under suitable conditions will form a duplex (helical) structure.

Estimating $e(s)$ terms





The change in free energy $e(s)$ is measured as a difference in "melting point" (highest temperature at which the molecule exists as a double-stranded region).

MFOLD

The free energy associated with loop regions can be estimated by constructions like the following:

$$\overbrace{G-G-\ldots-G-G}^{r}\underbrace{-A-A-\ldots-A-A-}_{m}\overbrace{C-C-\ldots-C-C}^{r}$$

The rGs will form a helix with the complementary strand rCs.
Vary m and measure the differences in melting temperature.
$\Rightarrow$ Similar experiments can be done for interior loops and bulges

MFOLD (cont.)

Volume 9 Number 1 1981 Nucleic Acids Research

Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information

Michael Zuker and Patrick Stiegler+

Division of Biological Sciences, National Research Council of Canada, Ottawa K1A 0R6, Canada

Zuker's Algorithm (simplified)

This algorithm finds the most thermodynamically stable secondary structure for a given RNA sequence.

$$W(i,j) = \min \begin{cases} W(i+1,j), \\ W(i,j-1), \\ V(i,j), \\ \min_{i \leq k < j}[W(i,k) + W(k+1,j)]. \end{cases}$$

where V models a segment such that i and j are paired,

$$V(i,j) = \min \begin{cases} V_1(i,j), & \text{hairpin closed by } i \bullet j \\ V_2(i,j), & \text{helix extension, bulge, interior loop} \\ V_3(i,j), & \text{multiple loop} \end{cases}$$

Note that V_2 involves searching through all possible loop lengths, in practice maximum loop lengths are imposed.

Zuker's Algorithm (simplified)

In the notation V_k, k is the order of the cycle.
V_2 models a helix extension, a bulge or an interior loop,

$$V_2(i,j) = \min_{i<i'<j'<j} [e(\text{motif}) + V(i',j')]$$

In practice, the size of a bulge or interior loop is often limited to 20 nucleotides or less.
V_3 models a multiple loop,

$$V_3(i,j) = \min_{i+1<k<j-1} [e(\text{motif}) + W(i+1,k) + W(k+1,j-1)]$$

In "real world" implementations, these equations contain many more cases to accurately model specific cases.

Implementations

- **MFOLD** and **RNAfold** are two well known implementations;

Implementations

- **MFOLD** and **RNAfold** are two well known implementations;
- Both **dot not** handle **pseudoknots**;

Implementations

- **MFOLD** and **RNAfold** are two well known implementations;
- Both **dot not** handle **pseudoknots**;
- Algorithm is in $\mathcal{O}(N^3)$;

Implementations

- **MFOLD** and **RNAfold** are two well known implementations;
- Both **dot not** handle **pseudoknots**;
- Algorithm is in $\mathcal{O}(N^3)$;
- **PKNOTS** is an implementation of the dynamic programming that includes pseudoknots;

Implementations

- **MFOLD** and **RNAfold** are two well known implementations;
- Both **dot not** handle **pseudoknots**;
- Algorithm is in $\mathcal{O}(N^3)$;
- **PKNOTS** is an implementation of the dynamic programming that includes pseudoknots;
- **PKNOTS** with pseudoknots is in $\mathcal{O}(N^6)$.

Performance of the Nearest-Neighbour Model (for a single sequence)

The nearest-neighbour model works reasonably well for small RNAs, **69** % and **71** % PPV (positive predictive value) for the tRNA and 5S rRNA, which are approximately 80 and 120 nucleotides long, respectively.
K. J. Doshi, J. J. Cannone C. W. Cobaugh, et R. R. Gutell (2004) Evaluation of the suitability of free-energy minimization using nearest-neighbor energy parameters for RNA secondary structure prediction. BMC Bioinformatics **5**(1):105.

How to **circumvent** these limitations?

- RNAs conserve secondary structure interactions more than they conserve their sequence;

How to **circumvent** these limitations?

- RNAs conserve secondary structure interactions more than they conserve their sequence;
- The nearest-neighbour model performs well on average but fails for certain sequences;

How to **circumvent** these limitations?

- RNAs conserve secondary structure interactions more than they conserve their sequence;
- The nearest-neighbour model performs well on average but fails for certain sequences;
- A multiple sequence alignment cannot be built using traditional approaches;

How to **circumvent** these limitations?

- RNAs conserve secondary structure interactions more than they conserve their sequence;
- The nearest-neighbour model performs well on average but fails for certain sequences;
- A multiple sequence alignment cannot be built using traditional approaches;
- The single-sequence approach and the sequence alignment problem can be merged into one problem;

How to **circumvent** these limitations?

- RNAs conserve secondary structure interactions more than they conserve their sequence;
- The nearest-neighbour model performs well on average but fails for certain sequences;
- A multiple sequence alignment cannot be built using traditional approaches;
- The single-sequence approach and the sequence alignment problem can be merged into one problem;
- As the number of input sequences increases it becomes unlikely that the nearest-neighbour model simultaneously fails for all of them.

eXtended Dynalign

- David Sankoff (1985) **Simultaneous solution of RNA folding, alignment and protosequence problems**. SIAM J. Appl. Math. **45**(5):810–825
- Objective function is a linear combination of the free energy of each sequence given the common secondary structure
- D.H. Mathews et D.H. Turner (2002) **Dynalign: An Algorithm for Finding the Secondary Structure Common to Two RNA Sequences.** J. Mol. Biol. **317**:191–203.
- We extended this work for three sequences.

Idea

```
Score= -578

GCCCGGGTGGTGTAGTGGCCCATCATACGACCCTGTCACGGTCG-TGACGCGGGTTCAAATCCCGCCTCGGGCGCCA
GTCGCAATGGTG-TAGTTGGGAGCATGACAGACTGAAGATCTGTTGGTCATCGGTTCGATCCCGGTTTGTGACACCA
GCCCCCAUCGUCUAGAGGCCUAGGACACCUCCCUUUCACGGAGG-CGACAGGGAUUCGAAUUCCCUUGGGGGUACCA

(((((((..(((..........))).(((((.......))))).....(((((.......)))))))))))....
```

Idea

- The objective function is a linear combination of the free energy of each sequence given the common structure

$$\Delta G^\circ_{\text{total}} = \Delta G^\circ_{\text{seq 1}} + \Delta G^\circ_{\text{seq 2}} + \Delta G^\circ_{\text{seq 3}} + \Delta G^\circ_{\text{insertions}}$$

- No terms for substitutions
- Solved by dynamic programming: constructing an alignment and a common secondary structure for $S_1[i,j]$, $S_2[k,l]$ and $S_3[m,n]$, from the smallest to the largest segment

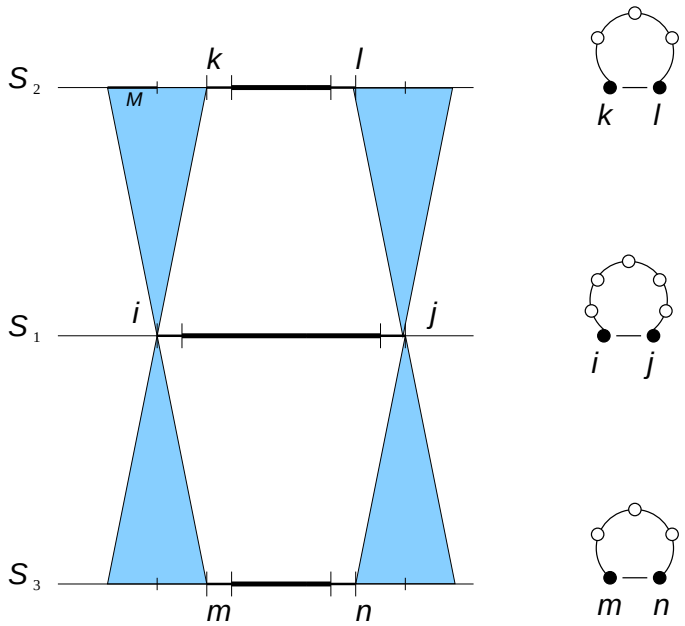
eXtended Dynalign

Let S_1, S_2 and S_3, be three RNA sequences.

- $W(i,j;k,l;m,n)$ represents the some of the free energy of $S_1[i,j]$, given the common structure, $S_2[k,l]$ given the common secondary structure and $S_3[m,n]$;
- $V(i,j;k,l;m,n)$ is defined similarly to W but also imposes constraints such that i is paired with j, k is paired with l, and m is paired with m;
- $W9$ represents the free energy for a prefix alignment of $S_1[1,j]$, $S_2[1,l]$ and $S_3[1,n]$.

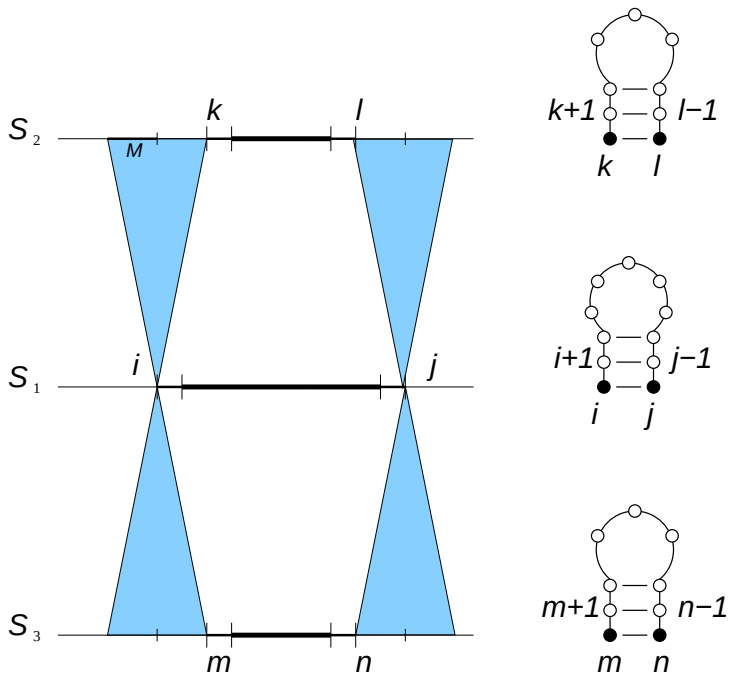
$\Rightarrow$ 140 cases: $V_1, V_2, V_{3_{1-64}}, W_1, W_2, W_{3_{1-64}}, W_{9_{1-8}}$.

Hairpin loop closed by a base-pair: $V_1(i,j;k,l;m,n)$



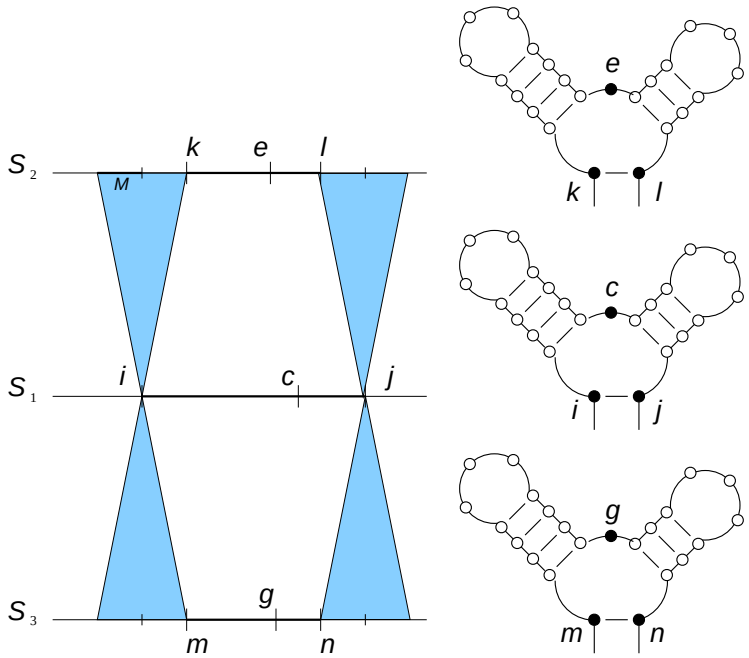
$$\Delta G^\circ_{\text{hairpin}}(i,j)+\Delta G^\circ_{\text{hairpin}}(k,l)+\Delta G^\circ_{\text{hairpin}}(m,n)+\Delta G^\circ_{\text{gap}}(\text{no. of gaps})$$

Helix Extension: $V_{2.1}(i,j;k,l;m,n)$



$$V(i+1,j-1;k+1,l-1;m+1,n-1)+\Delta G^{\circ}_{\text{motif}_1}+\Delta G^{\circ}_{\text{motif}_2}+\Delta G^{\circ}_{\text{motif}_3}$$

Multibranch Loop: $V_{3.1}(i,j;k,l;m,n)$



$$W(i,c;k,e;m,g)+W(c+1,j;e+1,l;g+1,n)+\Delta G^{\circ}_{\text{motif}_1}+\Delta G^{\circ}_{\text{motif}_2}+\Delta G^{\circ}_{\text{motif}_3}$$

tRNA Dataset

Id	Length	Description
RD0260	77	Asp Phage T5 (Virus)
RD0500	76	Asp *Haloferax volcanii* (Archae)
RD4800	71	Asp *Aedes albopictus* (Mitochondria, Animal)
RE2140	76	Glu *Synechocystis sp.* (Eubacteria)
RE6781	76	Glu *Hordeum vulgare* (Chloroplast)
RF6320	76	Phe *Schizosaccharomyces pombe* (Cytoplasm, Fungi)
RL0503	88	Leu *Haloferax volcanii* (Archae)
RL1141	89	Leu *Mycoplasma capricolum* (Eubacteria)
RS0380	88	Ser *Halobacterium cutirubrum* (Archae)
RS1141	92	Ser *Mycoplasma capricolum* (Eubacteria)

The percentage of sequence identify varies from 27.3 to 68.8 %.

MFOLD: tRNAs

Id	Sensitivity	PPV	MCC
RD0260	33.3	29.2	31.2
RD0500	47.6	43.5	45.5
RD4800	42.9	56.2	49.1
RE2140	95.2	87	91
RE6781	33.3	28	30.6
RF6320	0	0	0
RL0503	0	0	0
RL1141	40	43.5	41.7
RS0380	52	56.5	54.2
RS1141	19.2	25	21.9

5S rRNAs

Id	Length	Description
AJ131594	117	*Delftia acidovorans*
AJ251080	117	*Geobacillus stearothermophilus*
K02682	120	*Micrococcus luteus*
M10816	119	*Geobacillus stearothermophilus*
M16532	121	*Thermus sp.*
M25591	117	*Geobacillus stearothermophilus*
V00336	120	*Escherichia coli*
X02024	119	*Sporosarcina pasteurii*
X02627	120	*Agrobacterium tumefaciens*
X04585	119	*Rhodobacter capsulatus*
X08000	122	*Arthrobacter oxydans*
X08002	122	*Arthrobacter globiformis*

The percentage of identity varies from 47.2 to 88.2%.

MFOLD: 5S rRNAs

Id	Sensitivity	PPV	MCC
AJ131594	23.7	60	37.7
AJ251080	26.3	45.5	34.6
D11460	15.8	37.5	24.3
K02682	20.5	40	28.6
M10816	31.6	70.6	47.2
M16532	10.3	21.1	14.7
M25591	26.3	45.5	34.6
V00336	37.5	65.2	49.5
X02024	15.8	37.5	24.3
X02627	38.5	68.2	51.2
X04585	0	0	0
X08000	0	0	0
X08002	0	0	0

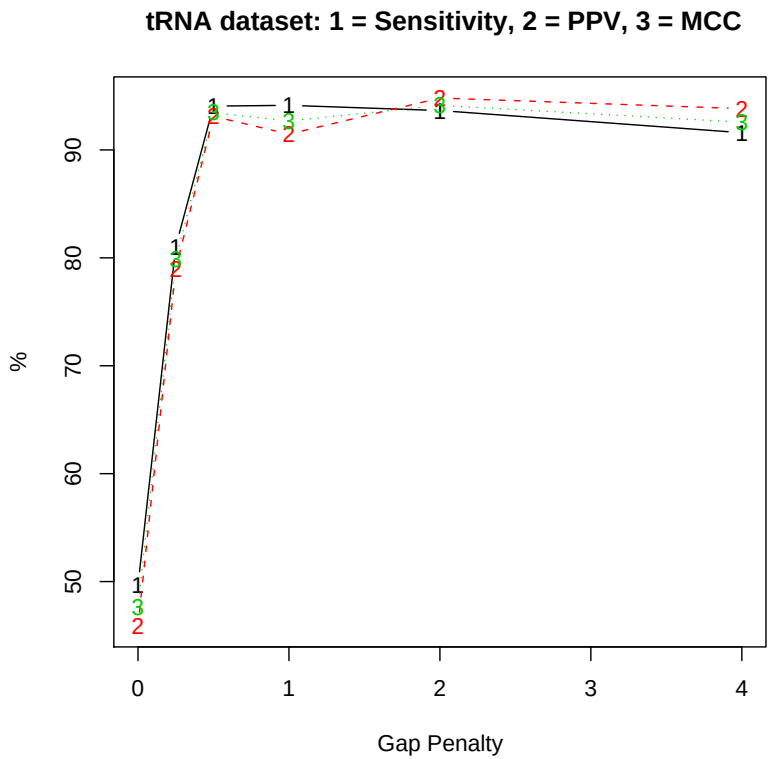
Are three input sequences better than two?

1. The worse prediction (minimum accuracy) should be more accurate;
2. Use of three input sequences should improve the average accuracy;
3. Average coverage should be less.

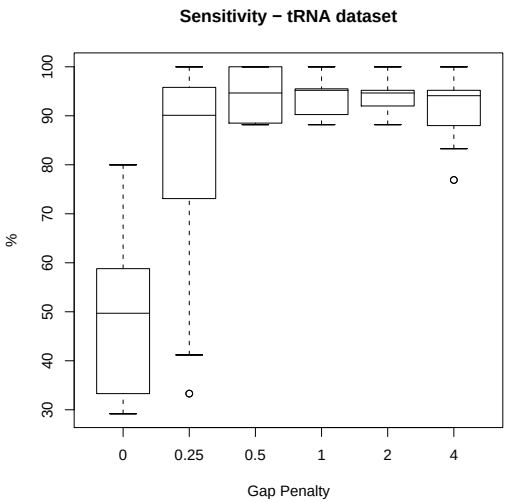
Masoumi, B. and Turcotte, M. (2005) Simultaneous alignment and structure prediction of three RNA sequences. *Int. J. Bioinformatics Research and Applications.* Vol. 1, No. 2, pp. 230-245

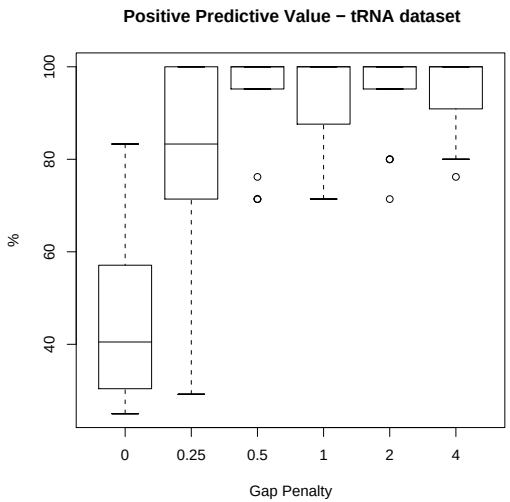
Beeta Masoumi and Marcel Turcotte. Simultaneous alignment and structure prediction of RNAs: Are three input sequences better than two? In S. V. Sunderam et al., editor, *2005 International Conference on Computational Science (ICCS 2005)*, Lecture Notes in Computer Science 3515, pages 936-943, Atlanta, USA, May 22-25 2005.

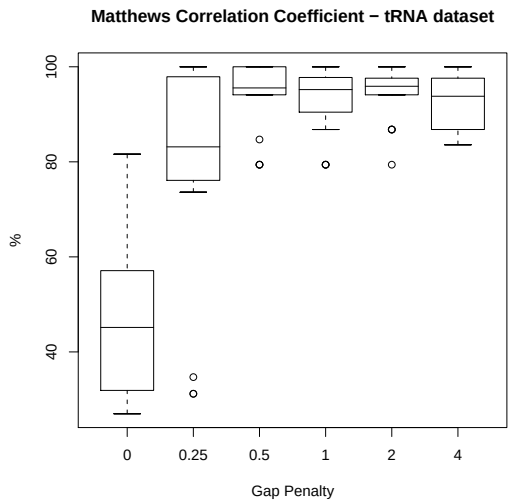
Calibrating Gap penalties: tRNAs



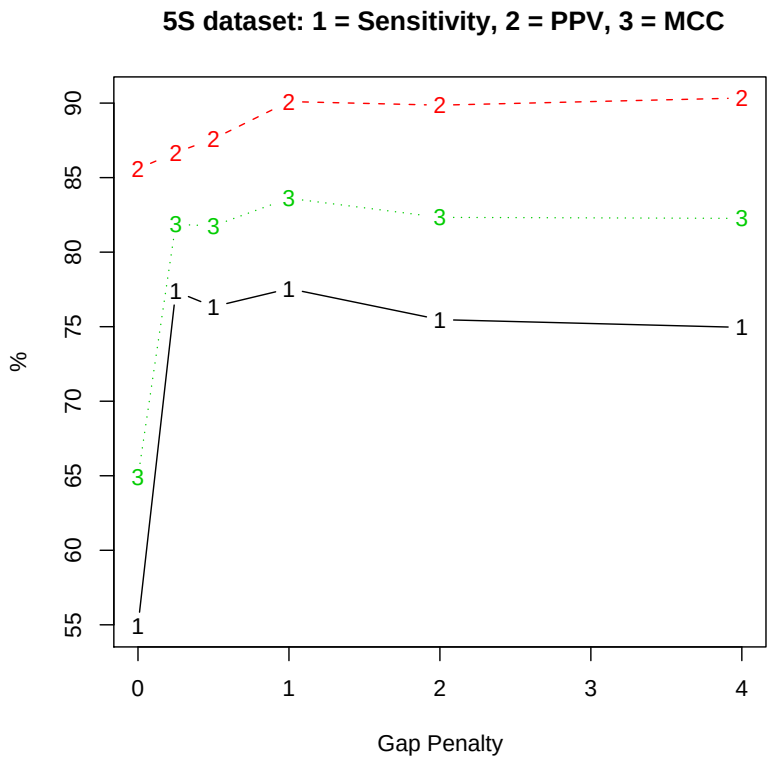
Calibrating Gap penalties: tRNAs



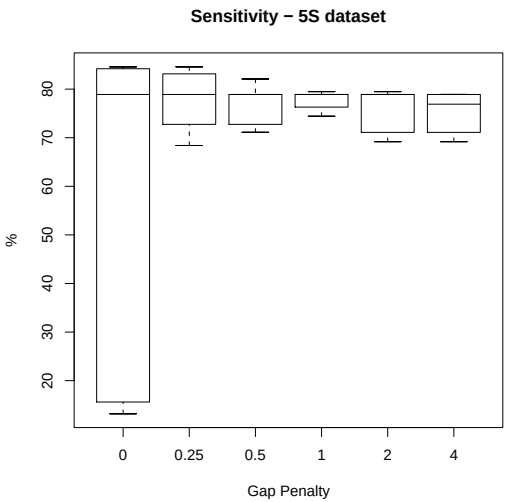


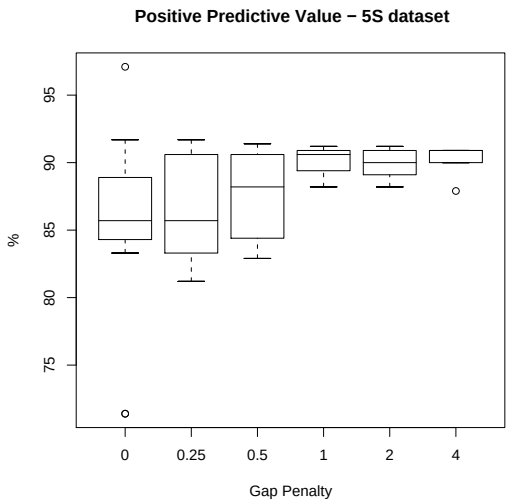


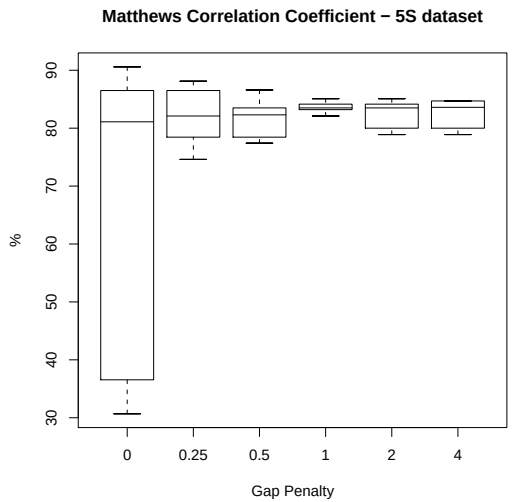
Calibrating Gap Penalties: 5S rRNAs



Calibrating Gap Penalties: 5S rRNAs







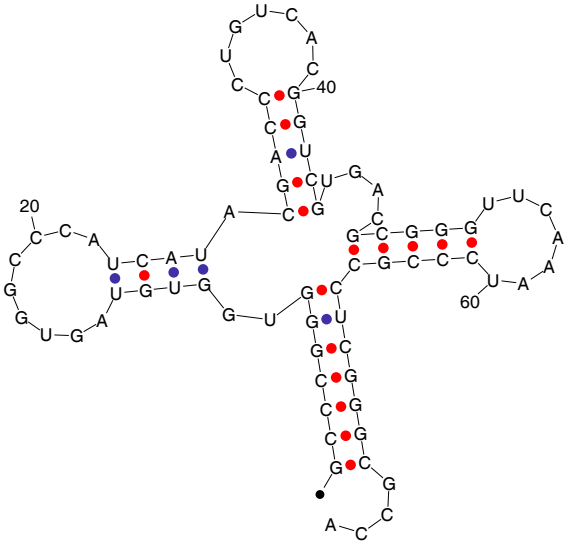
PPV: tRNA Dataset

Id	N_{xd}	N_d	Min_{xd}	Min_d	Max_{xd}	Max_d	Ave_{xd}	Ave_d
RD0260	4	5	100	80	100	100	100.0	96.0
RD0500	4	5	76	45	100	100	82.2	80.8
RD4800	5	5	100	80	100	100	100.0	96.0
RE2140	2	4	100	100	100	100	100.0	100.0
RE6781	2	4	100	77	100	100	100.0	94.3
RF6320	4	5	95	45	100	100	96.4	89.1
RL0503	1	2	100	100	100	100	100.0	100.0
RL1141	2	3	100	70	100	100	100.0	90.3
RS0380	1	2	100	83	100	87	100.0	85.2
RS1141	2	3	100	70	100	100	100.0	90.3

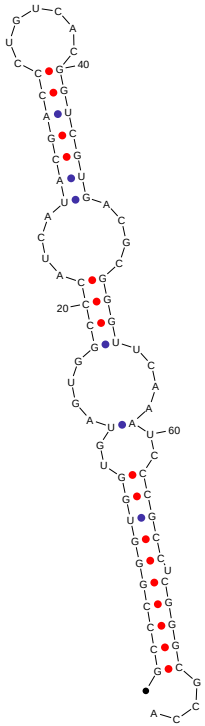
xd stands for eXtended Dynalign, *d* stands for Dynalign.

X-Dynalign 96.8 ± 7.6 vs Dynalign 92.1 ± 14.6.

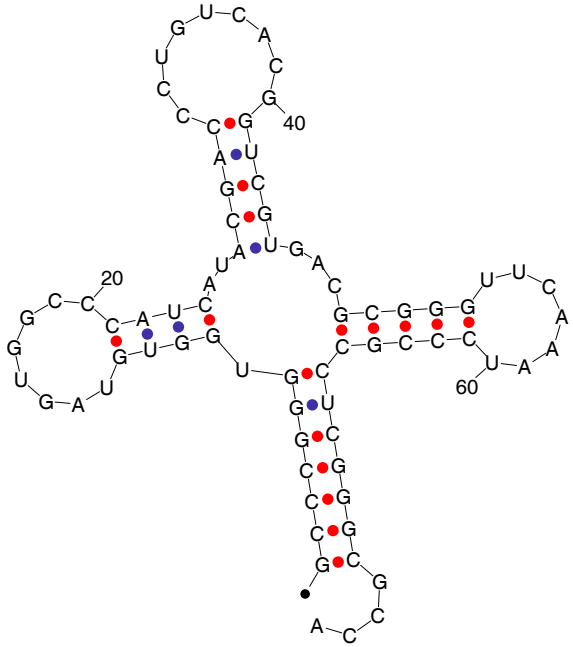
eXtended-Dynalign reproduces the clover-leaf structure



(a) RD0500

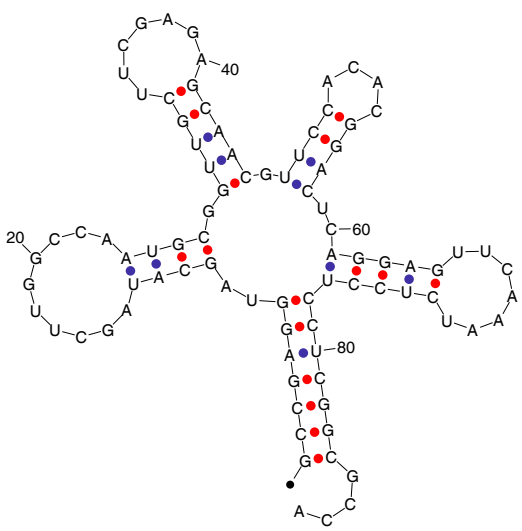


(b) Dynalign

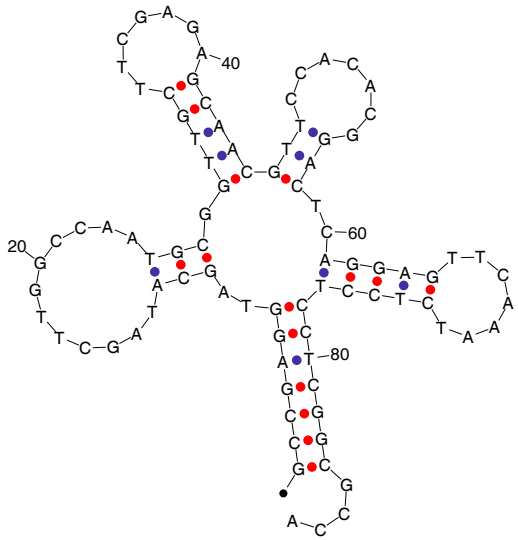


(c) X-Dynalign

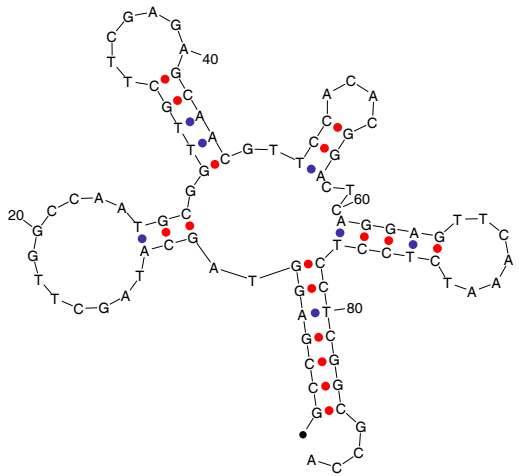
Fine details are better reproduced as well



(a) RS0380



(b) Dynalign



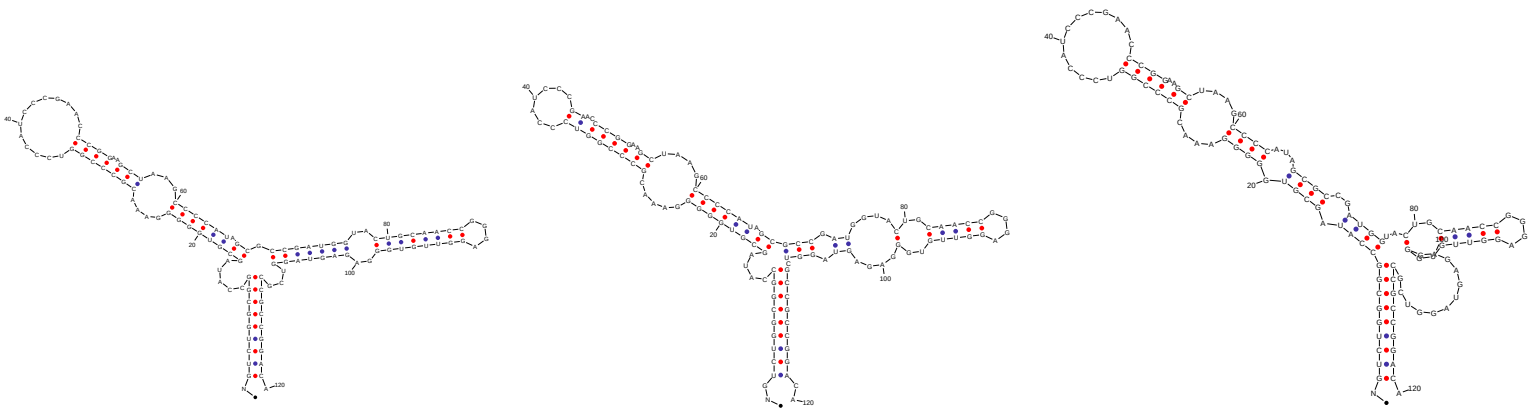
(c) X-Dynalign

PPV: 5S rRNA

Id	N_{xd}	N_d	Min_{xd}	Min_d	Max_{xd}	Max_d	Ave_{xd}	Ave_d
AJ131594	2	3	100	91	100	100	100.0	94.5
AJ251080	6	5	88	82	90	86	90.3	84.8
D11460	6	5	87	66	87	88	87.6	79.4
K02682	8	9	63	88	100	97	89.1	92.0
M10816	3	4	90	85	90	88	90.7	87.8
M16532	1	2	94	77	94	85	94.1	81.8
M25591	6	5	87	82	90	86	89.8	84.8
V00336	3	4	75	65	100	100	91.9	91.4
X02024	9	6	88	82	90	88	90.1	85.8
X02627	1	2	100	92	100	100	100.0	96.0
X04585	2	3	72	68	94	93	83.4	82.7
X08000	5	5	90	88	90	90	90.6	89.4
X08002	5	5	90	88	90	90	90.6	89.4

X-Dynalign 90.3 ± 5.8, Dynalign = 87.7 ± 7.4.

(K02682,V00336,X04585), PPV = 63%



Reference, Dynalign and X-Dynalign structures for the 5S rRNA K02682.

Pros: eXtended Dynalign

- The mean PPV is higher;
- Better worse case scenario;
- The average sensitivity is slightly degraded. However, for the majority of the sequences the minimum sensibility is higher for eXtended Dynalign;
- Some subtle details, such as the variable loop of some tRNAs, are well reproduced.

Cons: eXtended Dynalign

- $\mathcal{O}(|S_1|^2 M^4)$ space, $\mathcal{O}(|S_1|^3 M^6)$ time;
- Severe constraint M, $M \leq 6$;
- Up to two weeks of CPU time for some sequences[†];
- Length limited to some 150 nucleotides.

[†] Sun Fire V20z, AMD Opteron 2.2 GHz, Solaris 9

How to circumvent these limitations

- How to go beyond 3 sequences?

How to circumvent these limitations

- How to go beyond 3 sequences?
  Bellamy-Royds, A. B. & Turcotte, M. Can Clustal-style progressive pairwise alignment of multiple sequences be used in RNA secondary structure prediction? *BMC bioinformatics* **8**, 190 (2007).
- How to handle longer sequences?

How to circumvent these limitations

- How to go beyond 3 sequences?
  Bellamy-Royds, A. B. & Turcotte, M. Can Clustal-style progressive pairwise alignment of multiple sequences be used in RNA secondary structure prediction? *BMC bioinformatics* **8**, 190 (2007).
- How to handle longer sequences?
  Using a window-based approach to study the secondary structure landscape of HDV;

How to circumvent these limitations

- How to go beyond 3 sequences?
  Bellamy-Royds, A. B. & Turcotte, M. Can Clustal-style progressive pairwise alignment of multiple sequences be used in RNA secondary structure prediction? *BMC bioinformatics* **8**, 190 (2007).
- How to handle longer sequences?
  Using a window-based approach to study the secondary structure landscape of HDV;
  Developing tests for determining the likelihood of a structure.

Seed: Summary

- Novel approach for discovering consensus secondary structure motifs in unaligned RNA sequences;

Seed: Summary

- Novel approach for discovering consensus secondary structure motifs in unaligned RNA sequences;
- Exhaustive exploration of a space induced from a Seed sequence using minimum support constraints;

Seed: Summary

- Novel approach for discovering consensus secondary structure motifs in unaligned RNA sequences;
- Exhaustive exploration of a space induced from a Seed sequence using minimum support constraints;
- Uses suffix arrays for enumerating stems (first step of the motif inference algorithm);

Seed: Summary

- Novel approach for discovering consensus secondary structure motifs in unaligned RNA sequences;
- Exhaustive exploration of a space induced from a Seed sequence using minimum support constraints;
- Uses suffix arrays for enumerating stems (first step of the motif inference algorithm);
- Uses suffix arrays for efficiently matching RNA secondary structure motifs (pattern matcher).

Seed: Summary

- Novel approach for discovering consensus secondary structure motifs in unaligned RNA sequences;
- Exhaustive exploration of a space induced from a Seed sequence using minimum support constraints;
- Uses suffix arrays for enumerating stems (first step of the motif inference algorithm);
- Uses suffix arrays for efficiently matching RNA secondary structure motifs (pattern matcher).

This particular phase of the project focuses on the exploration of the search space. We are currently investigating objectives functions in a second phase.

Seed: Research objectives (1/2)

Developing a tool taking as input an ensemble of (unaligned) sequences and producing as output a list of conserved structural motifs.

Seed: Research objectives (1/2)

Developing a tool taking as input an ensemble of (unaligned) sequences and producing as output a list of conserved structural motifs.
With the following additional constraints:

- No (or little) sequence similarity;
- More than one family present in the input sequences.

Seed: Research objectives (2/2)

For this particular phase of the project, we wanted answers to the following questions.

- Are support and exclusion constraints sufficiently powerful to make an exhaustive search of the secondary structure space feasible?
- Does the search space contain biologically interesting motifs?

Mohammad Anwar, Truong Nguyen and Marcel Turcotte (2006) Identification of consensus RNA secondary structures using suffix arrays. BMC Bioinformatics, 7:244.

Truong Nguyen and Marcel Turcotte (2005) Exploring the Space of RNA Secondary Structure Motifs Using Suffix Arrays. 6th International Symposium on Computational Biology and Genome Informatics (CBGI 2005). Editors S. Blair et al., Salt Lake City, Utah, USA, July 21-26, 2005, 1291–1298.

Why proposing a new method?

We think that existing methods are not appropriate for studying regulatory motifs.

- Exact methods, such as eXtended Dynalign, are limited to 3 short sequences. Furthermore, the common secondary structure cannot be more than M positions apart;
- Less structured;
- Modular;
- Unaligned;

Overview (1/6)

- Input: k unaligned sequences;
- Select a **seed** sequence;
- Within the search space induced from the seed sequence report all the motifs that are matching a sufficiently large number of the input sequences (support).

Phase I focused on building an efficient framework for exploring the space of RNA secondary structure motifs.
Phase II (just started) will focus on building effective objective functions.

Overview (2/6)

```
>RD0260 (*)
GCGACCGGGGCUGGCUUGGUAAUGGUACUCCCCUGUCACGGGAGAGAAUGUGGGUUCAAAUCCCAUCGGTCGCGCCA
>RD0500
GCCCGGGUGGUGUAGUGGCCCAUCAUACGACCCUGUCACGGUCGUGACGCGGGUUCAAAUCCCGCCUCGGGCGCCA
>RD1140
GGCCCCAUAGCGAAGUUGGUUAUCGCGCCUCCCUGUCACGGAGGAGAUCACGGGUUCGAGUCCCGUUGGGGUCGCCA
>RD2640
GGGAUUGUAGUUCAAUUGGUCAGAGCACCGCCCUGUCAAGGCGGAAGAUGCGGGUUCGAGCCCCGUCAGUCCCGCCA
>RE2140
GCCCCCAUCGUCUAGAGGCCUAGGACACCUCCCUUUCACGGAGGCGACAGGGAUUCGAAUUCCCUUGGGGGUACCA
>RE6781
UCCGUCGUAGUCUAGGUGGUUAGGAUACUCGGCUCUCACCCGAGAGACCCGGGUUCGAGUCCCGGCGACGGAACCA
>RF6320
GUCGCAAUGGUGUAGUUGGGAGCAUGACAGACUGAAGAUCUGUUGGUCAUCGGUUCGAUCCCGGUUUGUGACACCA
```

In this example, there are 7 input sequences and RD0260 has been selected to be the Seed sequence.

Overview (3/6)

```
[ find_all_stems ]

GCGACCGGGGCTGGCTTGGTAATGGTACTCCCCTGTCACGGGAGAGAATGTGGGTTCAAATCCCATCGGTCGCGCCA

NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
(((....................................................................)))

NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
(((((((..........................................................)))))))

NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
(((............................................)))
...
                                                    NNNNNNNNNNNNNNNNN
                                                    (((...........)))

                                                          NNNNNNNNNNN
                                                          (((.....)))
```

Overview (4/6)

```
[ fix_all ]

GCGACCGGGGCTGGCTTGGTAATGGTACTCCCCCGCCACGGGAGAGAATGTGGGTTCAAATCCCATCGGTCGCGCCA

NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
(((.............................)))

NNGNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNCNN
(((.............................)))

NCNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNGN
(((.............................)))

GNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNC
(((.............................)))

GCNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNGC
(((.............................)))
...
```

Overview (5/6)

```
[ combine_all ]

GCGACCGGGGCTGGCTTGGTAATGGTACTCCCCTGTCACGGGAGAGAATGTGGGTTCAAATCCCATCGGTCGCGCCA

                    GNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNC
                    (((.....................................)))

                                        +

                         NNNTNNNNNNNNGNNN
                         (((((......)))))

                                        =

                    GNNNNNNNTNNNNNNNNGNNNNNNNNNNNNNNNNNNNNNNNNNNC
                    (((..(((((......)))))..................)))
```

The motifs with insufficient support are rejected.

Overview (6/6)

```
[ combine_all ]

GCGACCGGGGCTGGCTTGGTAATGGTACTCCCCTGTCACGGGAGAGAATGTGGGTTCAAATCCCATCGGTCGCGCCA

                                  CNTNNNNNGNG
                                  (((.....)))

                                        +

                                                  GNNTNNNNGNNC
                                                  ((((....))))

                                        =

                                  CNTNNNNNGNGNNNGNNTNNNNGNNC
                                  (((.....)))...((((....))))
```

Subsequently, the 3 helices motifs, 4 helices motifs …will be produced.

Observations

- Huge search space

Observations

- Huge search space
- **Support** and **exclusion** should be powerful constraints

Observations

- Huge search space
- **Support** and **exclusion** should be powerful constraints
- Motifs will be matched against a fix set of sequences (over and over again)

Observations

- Huge search space
- **Support** and **exclusion** should be powerful constraints
- Motifs will be matched against a fix set of sequences (over and over again)

Motif discovery framework

A motif discovery approach can be characterised by,

1. its search space
2. the algorithm that is used to search the space
3. its objective function

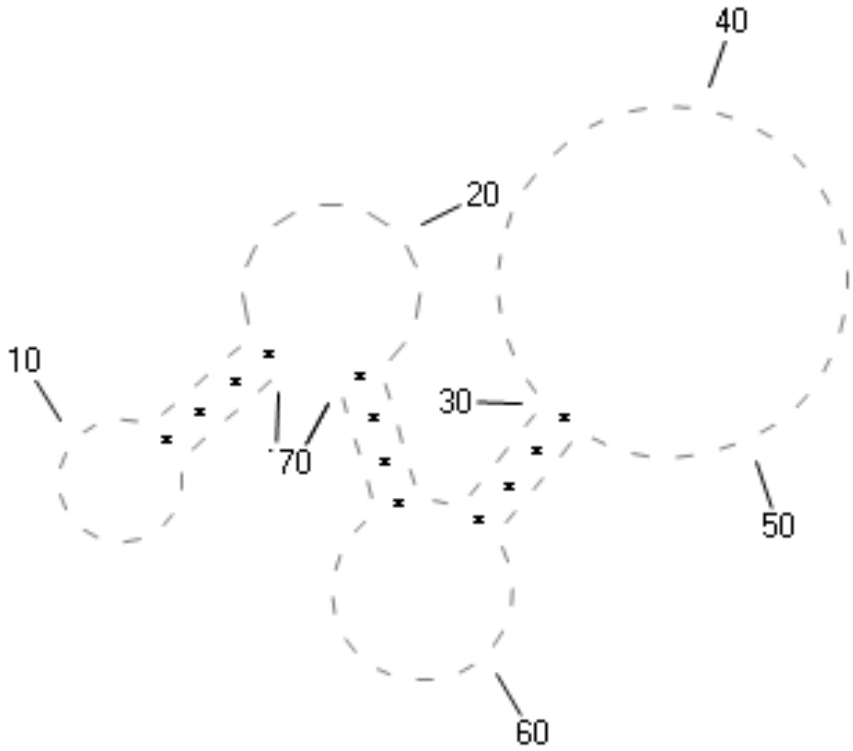
A. Brazma, I. Jonassen, I. Eidhammer et D. Gilbert (1998) *Journal of Computational Biology* 5:279-305.

Search space (1/2)

Let $\Sigma = \{A, C, G, T\} \cup \{N, N'\}$, the nucleotides alphabet augment with the joker symbols N and N', where $N \in \{A, C, G, T\}$ and N' is its complement.
The notation $5 : E$ represents the 5' end of a paired region, and E is a word on Σ.
The notation $3 : E$ represents the 3' end of a paired region, and E is a word on Σ.
The notation $D : n$ represents a distance constraint.

```
5:CNGA D:7 3:TCN'G D:7 5:NNGG D:0 5:NAAG D:23 3:CTTN' D:
```

Search space (2/2)



5:CNGA D:7 3:TCN’G D:7 5:NNGG D:0 5:NAAG D:23 3:CTTN’ D:

Seed: Search algorithm

Sequential covering

1. **while** there are more examples

Seed: Search algorithm

Sequential covering

1. **while** there are more examples
   1.1 select an example randomly (Seed sequence)

Seed: Search algorithm

Sequential covering

1. **while** there are more examples
   1.1 select an example randomly (Seed sequence)
   1.2 build the most specific motif

Seed: Search algorithm

Sequential covering

1. **while** there are more examples
   - 1.1 select an example randomly (Seed sequence)
   - 1.2 build the most specific motif
   - 1.3 general-to-specific search

Seed: Search algorithm

Sequential covering

1. **while** there are more examples
   - 1.1 select an example randomly (Seed sequence)
   - 1.2 build the most specific motif
   - 1.3 general-to-specific search
   - 1.4 remove all the examples containing an instance of the "best" motif

Seed: Search algorithm

Sequential covering

1. **while** there are more examples
   - 1.1 select an example randomly (Seed sequence)
   - 1.2 build the most specific motif
   - 1.3 general-to-specific search
   - 1.4 remove all the examples containing an instance of the "best" motif

The number of families of motifs present in the input sequences is not know *a priori*, this kind of algorithm "may" help uncover this number.

Building the most specific motif (1/6)

- Let a be the Seed sequence picked up randomly at the previous step.

Building the most specific motif (1/6)

- Let a be the Seed sequence picked up randomly at the previous step.
- All the complementary regions of some minimum length, possibly containing GU base pairs and mismatches are enumerated.

Building the most specific motif (1/6)

- Let a be the Seed sequence picked up randomly at the previous step.
- All the complementary regions of some minimum length, possibly containing GU base pairs and mismatches are enumerated.
- Conceptually done using suffix trees and LCA (Lowest Common Ancestor) — in practice, suffix arrays have been used.

Building the most specific motif (2/6)

- Let S^c denote the complement of S; i.e. a string of the same length as S where all the As have been replaced by Us, all the Cs by Gs, all the Gs by Cs and all the Us by As, e.g. AACGU is the complement of UUGCA.

Building the most specific motif (2/6)

- Let S^c denote the complement of S; i.e. a string of the same length as S where all the As have been replaced by Us, all the Cs by Gs, all the Gs by Cs and all the Us by As, e.g. AACGU is the complement of UUGCA.
- Let S^r denote the reverse of S; i.e. this is the string S written backwards, e.g. UGCAA is the reverse of AACGU.

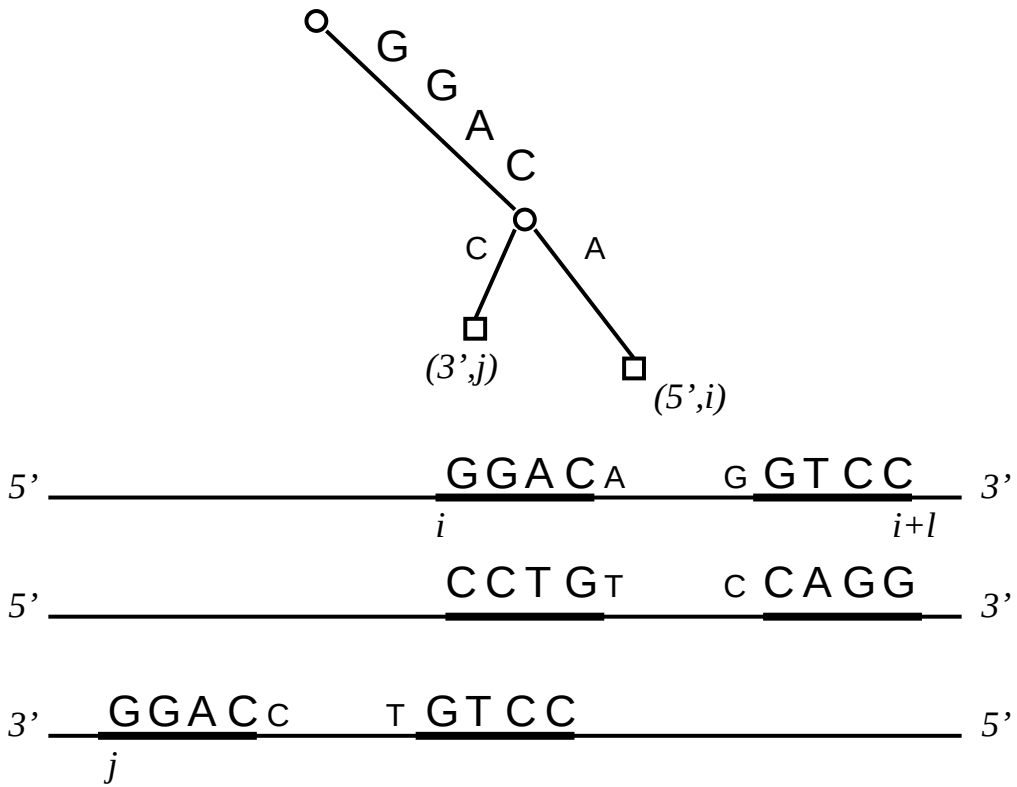
Building the most specific motif (2/6)

- Let S^c denote the complement of S; i.e. a string of the same length as S where all the As have been replaced by Us, all the Cs by Gs, all the Gs by Cs and all the Us by As, e.g. AACGU is the complement of UUGCA.
- Let S^r denote the reverse of S; i.e. this is the string S written backwards, e.g. UGCAA is the reverse of AACGU.
- Let S^{rc} be the reverse complement of S; e.g. ACGUU is the reverse complement of AACGU. A pair of strings (S, S^{rc}) is called a biological palindrome.

Building the most specific motif (3/6)

Proposition. Given i and j, determining the largest k such that $S[i, i+k-1]$ and $S[j-k+1, j]$ forms a biological palindrome can be done in constant time.

Building the most specific motif (4/6)



$\Rightarrow$ where $j = |S| - i - l + 1$.

Building the most specific motif (5/6)

```
GCGGCGGTGGCTGGCTTGGTAATGAGCAACCGTCGCCACGGGAGAGAATGTGGGTTCAAATC
(((X............................X)))
i                                 j
    ((((X................X))))      -> GU base pair
    i                       j
         (((X.........X)))            -> Mismatch
         i               j

Output

((((((((((((.............))))))))))))
```

Building the most specific motif 6/6)

1. Build a generalised suffix tree for S and $\overline{S}$ — where $\overline{S}$ is the reverse complement of S;
2. Annotate the tree for LCA queries;
3. **For** $i = 1 \ldots |S|$
   - 3.1 **For** $l = c_1 \ldots c_2$
     - 3.1.1 $j = |S| - i - l + 1$
     - 3.1.2 **If** $\text{LCA}((5', i), (3', j)) \geq c_1$ a complementary region has been found.

$\Rightarrow$ The actual algorithm has an additional inner loop allowing for GU base pairs and up to k mismatches.

Exploring the space of sec struc motifs (1/3)

The result of the previous step is a list of biological palindromes for the sequence S — this list possibly contains many conflicting stems.

Exploring the space of sec struc motifs (1/3)

The result of the previous step is a list of biological palindromes for the sequence S — this list possibly contains many conflicting stems. Each palindrome is first transformed into a generic (structural) motif — all the base pairs are replaced by $N \cdot N'$.

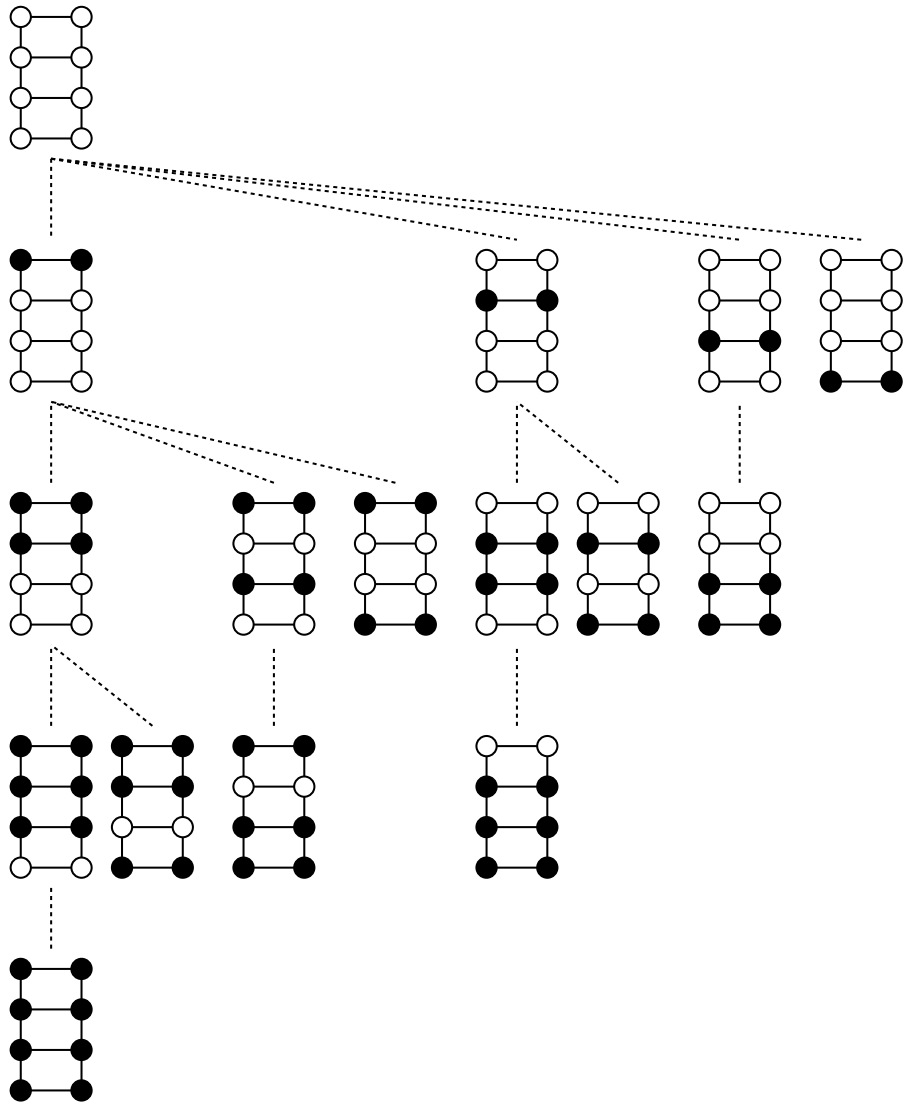
Exploring the space of sec struc motifs (1/3)

The result of the previous step is a list of biological palindromes for the sequence S — this list possibly contains many conflicting stems.
Each palindrome is first transformed into a generic (structural) motif — all the base pairs are replaced by $N \cdot N'$.
Conceptually, the root of the search tree is ϵ, the empty motif.

Exploring the space of sec struc motifs (1/3)

The result of the previous step is a list of biological palindromes for the sequence S — this list possibly contains many conflicting stems.
Each palindrome is first transformed into a generic (structural) motif — all the base pairs are replaced by $N \cdot N'$.
Conceptually, the root of the search tree is ϵ, the empty motif.
Two operations allow to create new motifs (nodes in the search tree): **instantiate** and **combine**.

Instantiate consists of replacing an $N \cdot N'$ base pair by the specific base pair that occurs at that position in the Seed sequence.



Combine creates a new motif by merging two existing motifs. There are two ways to combine motifs, nested or adjacent.
Combining nested motifs

```
  1. 5:NNGG D:40 3:CCN'N'
+
  2. 5:NAAG D:23 3:CTTN'
```

Produces

```
5:NNGG D:0 5:NAAG D:23 3:CTTN' D:9 3:CCN'N'
[    1     [         2         ]      1     ]
```

Combine creates a new motif by merging two existing motifs. There are two ways to combine motifs, nested or adjacent.
Combining adjacent motifs

```
  1. 5:CNGA D:7 3:TCN'G
+
  2. 5:NNGG D:40 3:CCN'N'
```

Produces

```
5:CNGA D:7 3:TCN'G D:7 5:NNGG D:40 3:CCN'N'
[       1         ]     [       2          ]
```

Exploring the space of sec struc motifs (2/3)

By construction, each newly created motif has at least one occurrence.

Exploring the space of sec struc motifs (2/3)

By construction, each newly created motif has at least one occurrence.
For each motif, let's define the **support** as the fraction of the k input sequences containing at least one occurrence of this motif.

Exploring the space of sec struc motifs (2/3)

By construction, each newly created motif has at least one occurrence.
For each motif, let's define the **support** as the fraction of the k input sequences containing at least one occurrence of this motif.
Any node (motif), and its subtree, is eliminated from the search space if its support is below a user specified threshold, typically 70%.

Exploring the space of sec struc motifs (2/3)

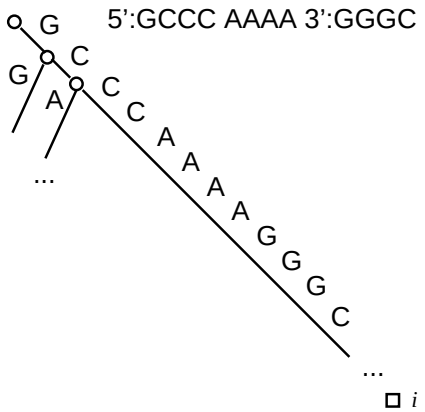
By construction, each newly created motif has at least one occurrence.
For each motif, let's define the **support** as the fraction of the k input sequences containing at least one occurrence of this motif.
Any node (motif), and its subtree, is eliminated from the search space if its support is below a user specified threshold, typically 70%.
The current strategy to explore the tree is a breath-first search — this allows the user to exhaustively explore the space of all the motifs containing 1,2,3, etc. stems.

Calculating the support: matching motifs (1/5)

- We have developed a non-deterministic algorithm inspired by Baeza-Yates & Gonnet's algorithm for regular expression matching.
  Baeza-Yates RA et Gonnet GH (1996) *J of the ACM* **43**(6):915–936.
- Given an input sequence b, $b \neq a$.
- The algorithm simultaneously traverses the suffix tree of b and the motif.
- It uses two stacks: the system stack is used to store backtracking points, while an explicit stack is used to validate matching elements of a base pair.

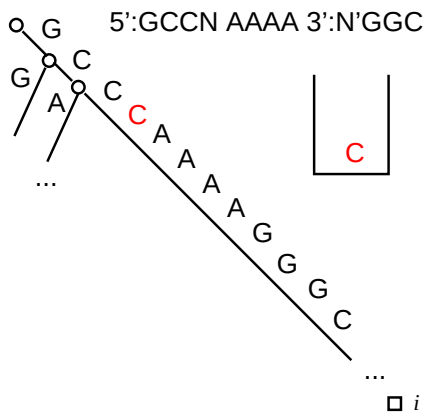
Calculating the support: matching motifs (2/5)

A symbol $\{A, C, G, T\}$ of the motif corresponds to itself in the suffix tree.
Matching a fully instantiated motif can be done in linear time w.r.t. the size of the motif.



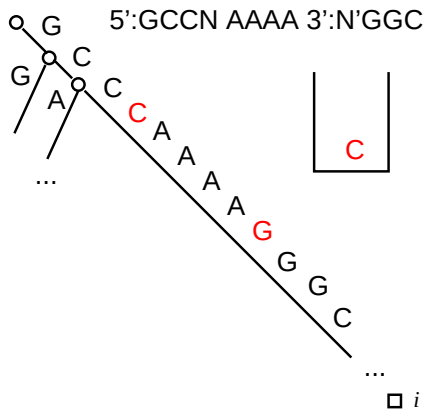
Calculating the support: matching motifs (3/5)

When a joker, *N*, is found in the 5' end of a stem region, and the last match occurred inside a label, the next character on that branch is pushed onto the base pair stack.



Calculating the support: matching motifs (4/5)

When a joker, N', is found in the 3' end of a stem region, the algorithm succeeds only if the next matching character can form a base pair with the element that is found on the top of the stack, then the top element is removed, and the algorithm continues.



Calculating the support: matching motifs (5/5)

Finally, when a joker is found, N, and the algorithm had stop at an interior node, all the branches are explored recursively (source of non-determinism).
Expressions such as this one, $D : n$, are processed similarly.

Exploring the space of sec struc motifs (3/3)

The algorithm stops if there are no more valid open nodes, or a user defined stopping criteria stops the algorithm.
The motifs are ranked and returned to the user.

Objective function(s)

$$\text{TSum} = \boxtimes_i \boxtimes_j \text{MFE}(m_{ij})$$

$$\text{TBest} = \boxtimes_i \min_j \text{MFE}(m_{ij})$$

$$\text{TWorst} = \boxtimes_i \max_j \text{MFE}(m_{ij})$$

where m_{ij} is the jth occurrence (match) in the ith sequence. We also defined variants of these functions where the free energy of a match is normalised by the number of base pairs.
Finally, we also used a simple objective function defined as the information content of the motif.

Implementation (1/3)

Suffix arrays are used rather than suffix trees.
Given an input sequence S of length $|S| = n$.
Each suffix is represented by its starting position (an integer), a suffix array lists all the suffixes in lexicographic order.
Uses $\mathcal{O}(n)$ space; with small constant.
$\log_2 n$ bits suffice to represent a position, hence, 32 bits, $4 \times n$ bits, are enough to represent a 4 Gbytes string.

Implementation (2/3)

Manber U et Myers G (1990) *Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms*: 319 – 327.
Manber U and Myers G (1993) *SIAM J on Computing* **22**(5):935–948.
Until very recently constructing a suffix array was costly, $\mathcal{O}(n \log n)$.
Building in $\mathcal{O}(n)$ time.
Kärkkäinen J et Sanders P (2003) In *Proc. 30th International Colloquium on Automata, Languages and Programming (ICALP '03), LNCS 2719*, 943-955. (Skew algorithm)

Implementation (3/3)

Bottom up traversal,
Abouelhoda M et al. (2003) WABI 2002, *LNCS 2452* :449-463.
Top down traversal,
Abouelhoda M et al. (2002) SPIRE 2002, *LNCS 2476* :31-43.
See Abouelhoda et al for an excellent review.
Mohamed Ibrahim Abouelhoda and Stefan Kurtz Enno Ohlebusch Replacing suffix trees with enhanced suffix arrays (2004) J. of Discrete Algorithms **(2):1**, 53–86.
16,000+ lines of C (now shrinked to some 8,000 lines).

Timing Results: Matcher (1/2)

Hordeum vulgare tRNA^{Glu} (RE6781)

```
> Full
UCCGUCGUAGUCUAGGUGGUUAGGAUACUCGGCUCUCACCCGAGAGACCCGGGUUCGAGUCCCGGCGACGGAACCA
((((((..((((.........)))).((((.......)))))....((((.......))))))))))....

> N_Loop
UCCGUCGNNGUCUNNNNNNNNNNGGAUNCUCGGNNNNNNNCCGAGNNNNCCGGGNNNNNNNCCCGGCGACGGANNNN
((((((..((((.........)))).((((.......)))))....((((.......))))))))))....

> N_Stem
NNNNNNNUANNNNAGGUGGUUANNNNANNNNNCUCUCACNNNNNAGACNNNNNUUCGAGUNNNNNNNNNNNNNACCA
((((((..((((.........)))).((((.......)))))....((((.......))))))))))....

> Generic
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
((((((..((((.........)))).((((.......)))))....((((.......))))))))))....
```

Find all matches allowing 1 mismatch in *Bacillus anthracis* 5,227,293 bp

Timing Results: Matcher (2/2)

# Nuc	Vtree	Full	N_Loop	N_Stem	Generic
512,000	0.7	0.000196	0.000270	0.37	0.80
1,024,000	2.0	0.000198	0.000379	0.68	1.67
2,048,000	4.6	0.000239	0.000618	1.32	3.56
4,096,000	10.3	0.000275	0.001201	2.49	7.75

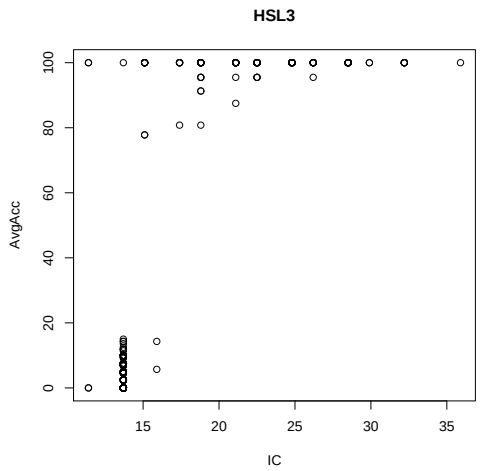
Times in seconds on Sun Fire V20z 2 $\times$ AMD Opteron 248 (2.2 GHz), 8 Gb, Solaris 9 (a single processor was used).

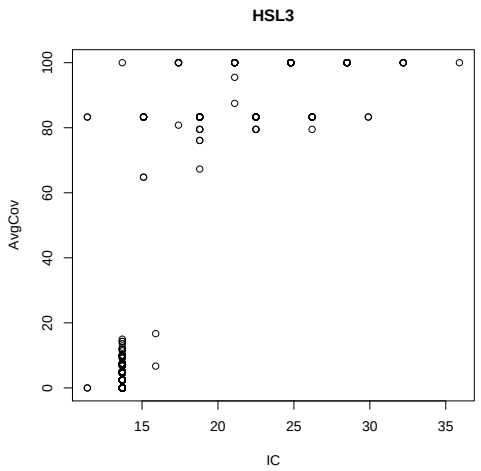
HSL302

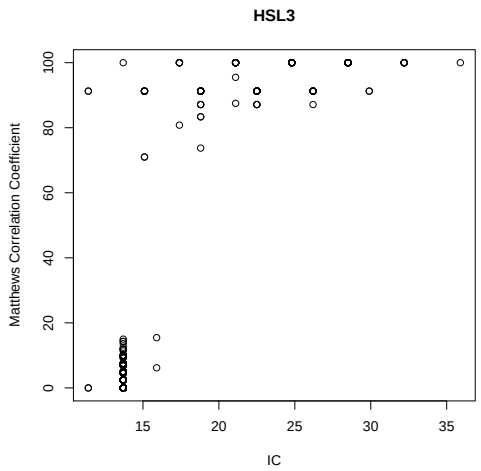
```
((((((....))))))
GGYYYTHHUHARRRCC
```

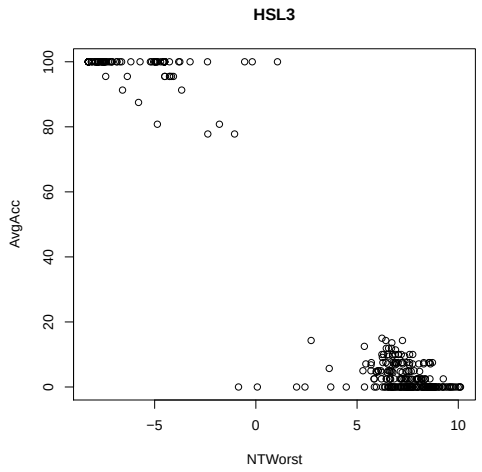
# Sequences	Length	# Motifs	# Matches	Space	Time
28	51–1,955	357	1,945,328	1.37 Mbytes	5m 21s

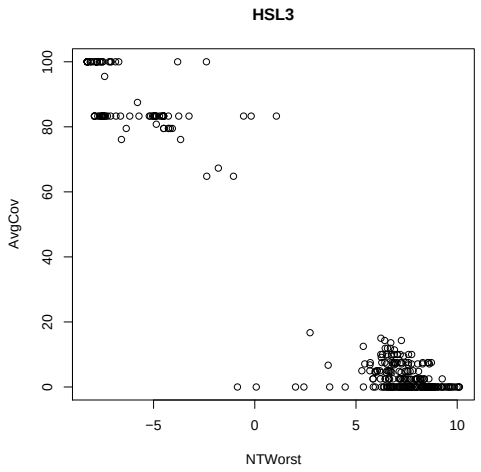
HSL302 (cont.)

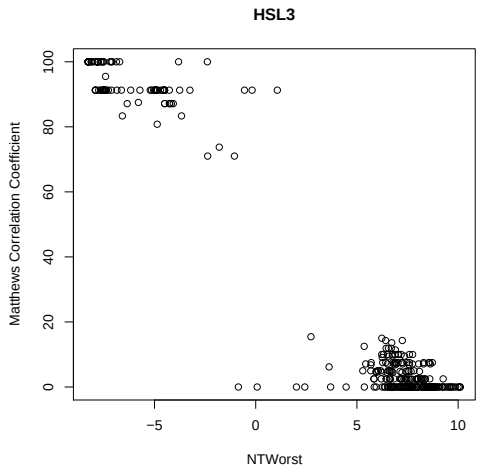

HSL3
AvgAcc
IC


HSL3
AvgCov
IC


HSL3
Matthews Correlation Coefficient
IC


HSL3
AvgAcc
NTWorst


HSL3
AvgCov
NTWorst


HSL3
Matthews Correlation Coefficient
NTWorst

HSL3 (motif 000269)

```
GGCNCTNNNNAGNGCC
((((((....))))))

((((((....))))))
GGYYYTHHUHARRRCC
```

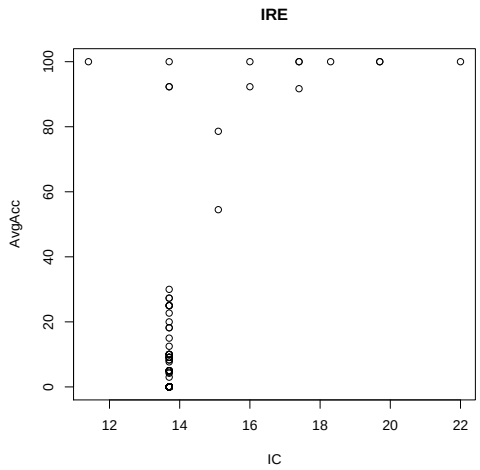
A third of the motifs inferred are 100 % accurate.

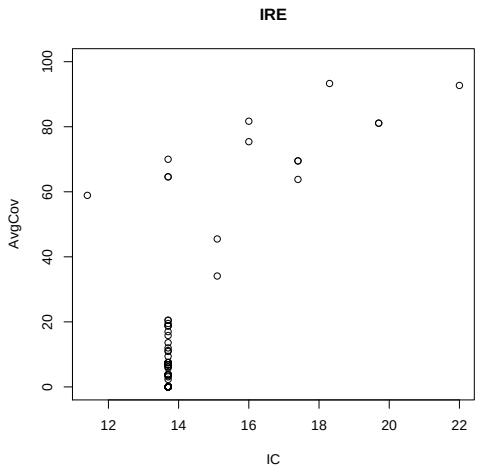
IRE

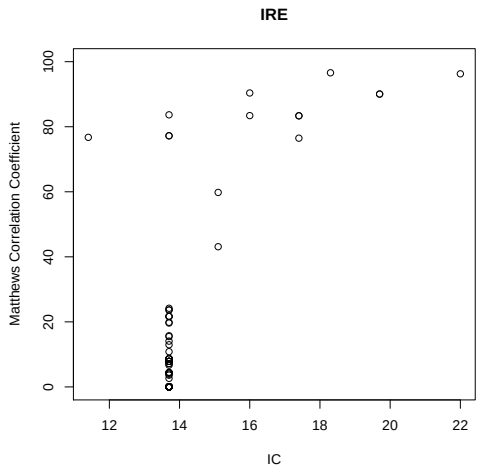
```
NNNCNNNNNCAGWGHNNNNNNNNN
(((.(((((.....)))))))))
```

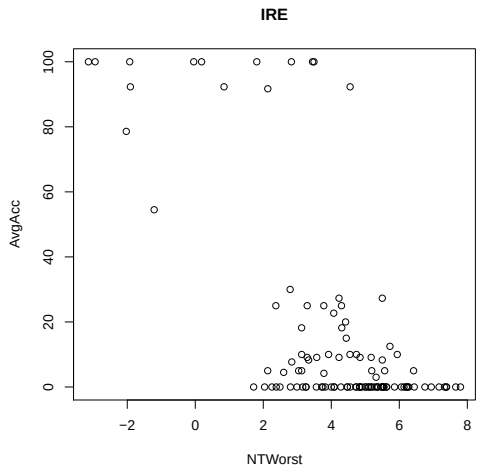
# Sequences	Length	# Motifs	# Matches	Space	Time
14	58–2,188	110	167,076	0.46 Mbytes	25s

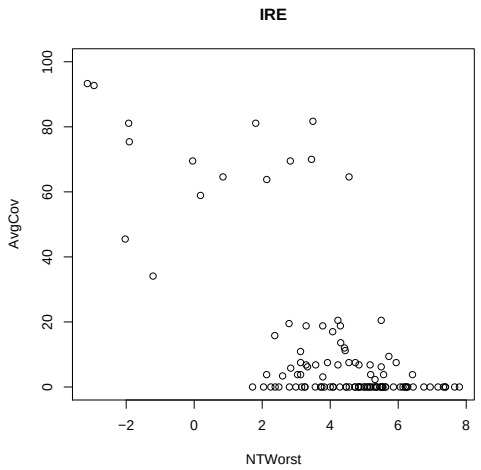
IRE (cont.)

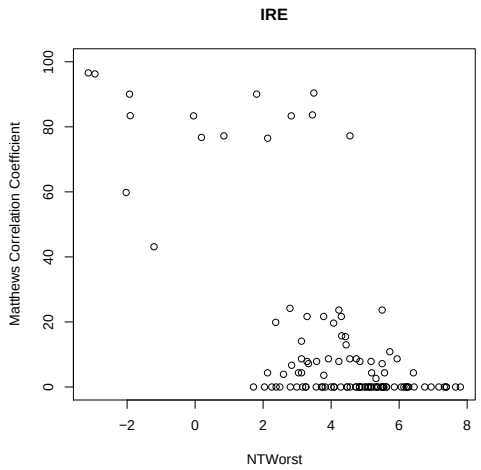

IRE
AvgAcc
IC


IRE
AvgCov
IC


IRE
Matthews Correlation Coefficient
IC


IRE
AvgAcc
NTWorst


IRE
AvgCov
NTWorst


IRE
Matthews Correlation Coefficient
NTWorst

IRE (motif 000086)

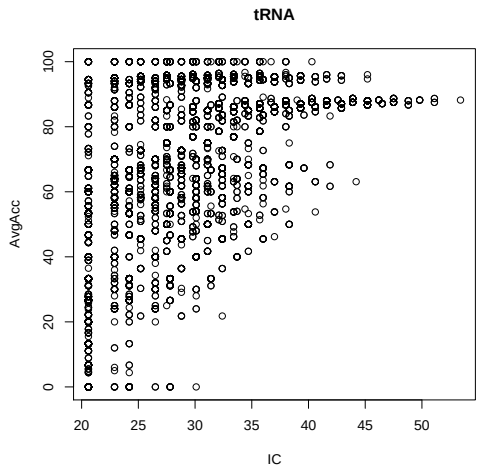
```
NNNNNNNNNNNNNNNNNNNNNNN
(((.(((((.....)))))))))

NNNCNNNNNCAGWGHNNNNNNNN
(((.(((((.....)))))))))
```

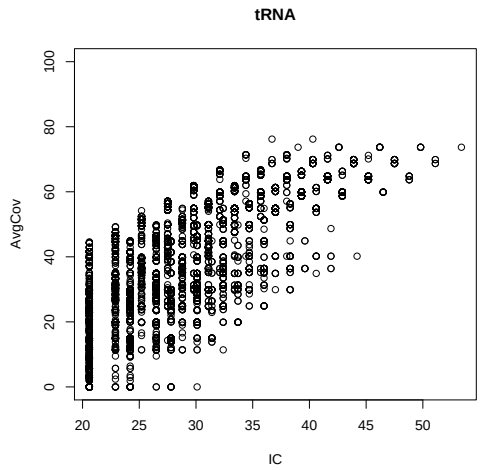
tRNA

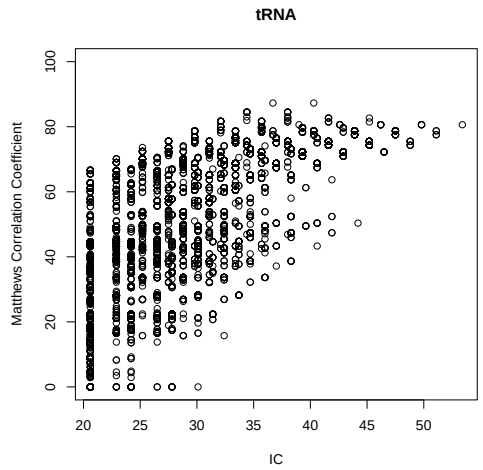
((((((..(((((.........)))))..(((((.......)))))).....(((((.......)

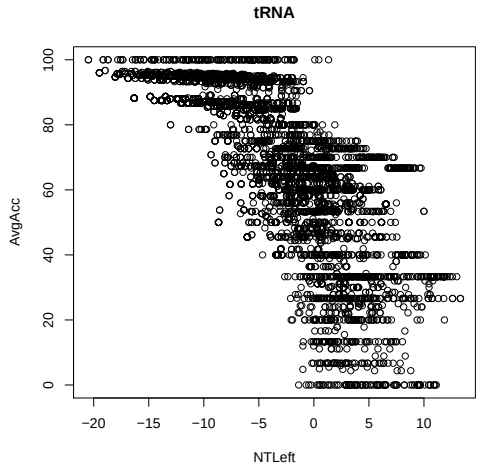
# Sequences	Length	# Motifs	# Matches	Space	Time
7	76–77	5,518	3,407,012	9.40 Mbytes	6m 11s

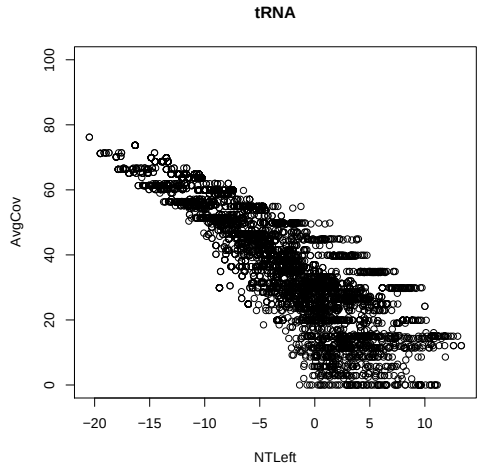
tRNA (cont.)

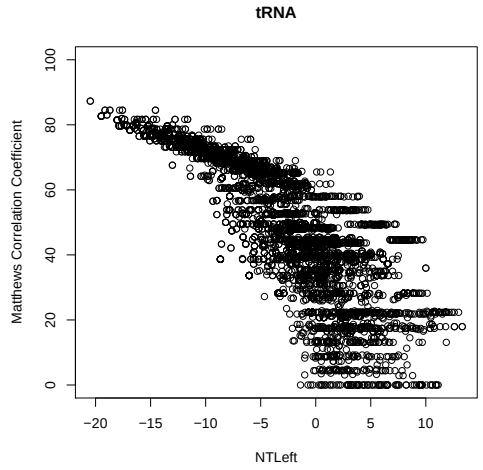

tRNA
AvgAcc
IC


tRNA
AvgCov
IC

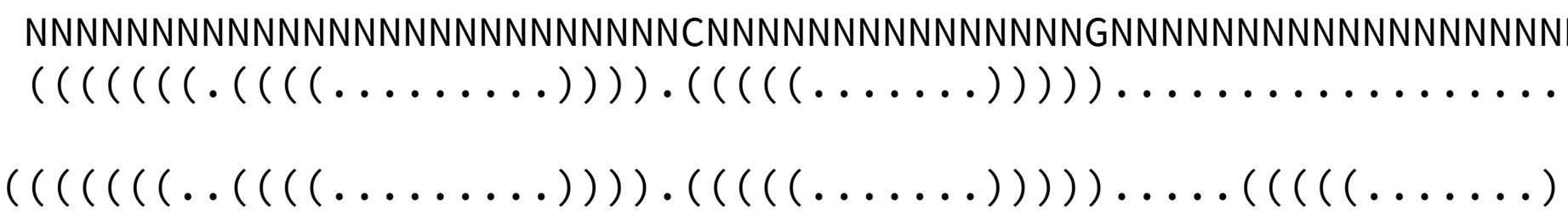

tRNA
Matthews Correlation Coefficient
IC


tRNA
AvgAcc
NTLeft

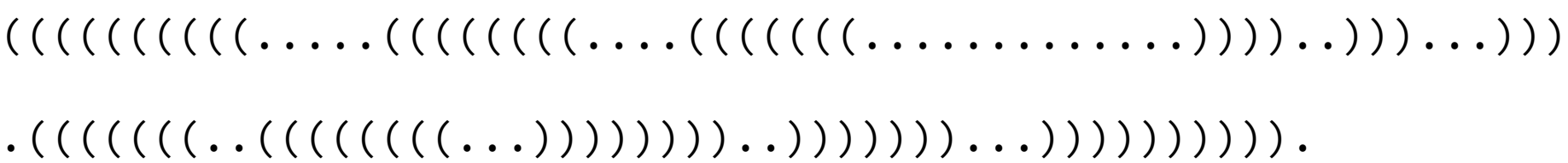

tRNA
AvgCov
NTLeft


tRNA
Matthews Correlation Coefficient
NTLeft

tRNA

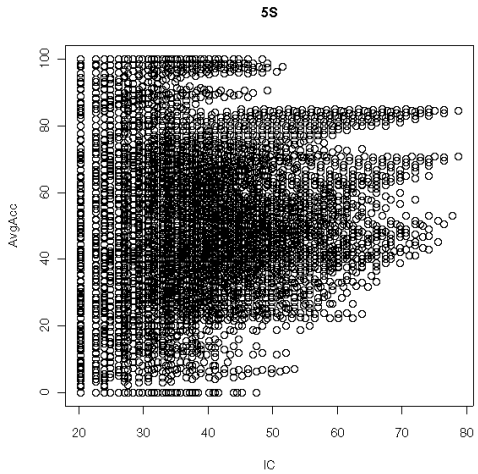

NNNNNNNNNNNNNNNNNNNNNNNNCNNNNNNNNNNNNNNNGNNNNNNNNNNNNNNNN
((((((((.((((.........)))).((((.......)))))).....................
((((((((..((((.........)))).((((.......)))))).....((((.......)))

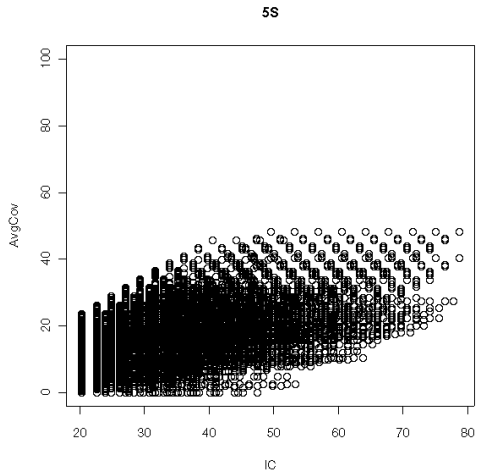
5S

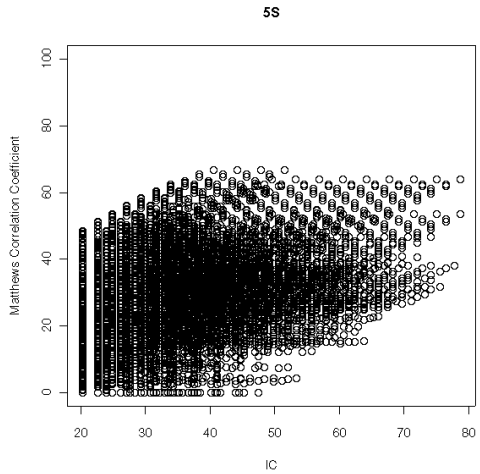


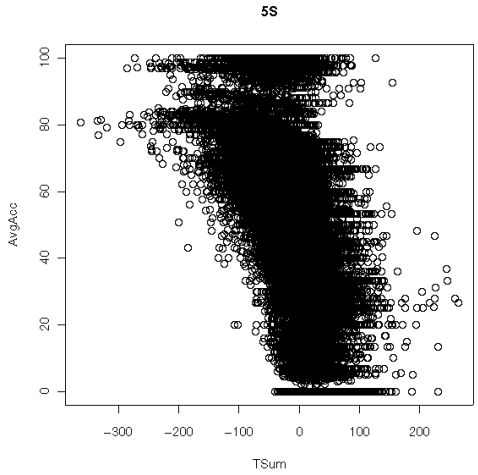
# Sequences	Length	# Motifs	# Matches	Space	Time
7	117–120	364,505	152,741,463	0.52 Gbytes	7h 40m

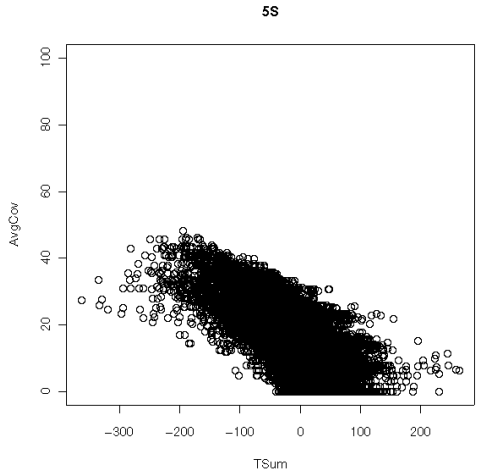
5S (cont.)

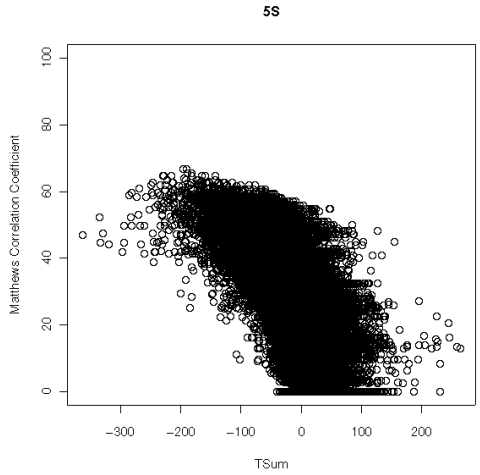

5S
AvgAcc
IC


5S
AvgCov
IC


5S
Matthews Correlation Coefficient
IC


5S
AvgAcc
TSum


5S
AvgCov
TSum


5S
Matthews Correlation Coefficient
TSum

5S

```
 NNNNGNNNNNNNNNNNNNNNNNNNNNNNNCNGNNNNNNNNNNNNNNNCNGNNNNNNNNNNN
 (((((((((.....................(((.............)))).............

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

NNNNNNNNNNNNNGNNNNNNNNNCNNNNNNNNNNNNNNNNNNNNCNNNN
.............(((.....)))..................)))))))

.(((((((..(((((((((...)))))))))..)))))))...)))))))))).
```

Publications

Mohammad Anwar and Marcel Turcotte (2006) An approach to selecting putative RNA motifs using MDL principle. *BIOCOMP'06 - The 2006 International Conference on Bioinformatics & Computational Biology*, pages 560-565, Las Vegas, Nevada, USA, June 26-29, 2006.

Mohammad Anwar and Marcel Turcotte (2006) Evaluation of RNA secondary structure motifs using regression analysis. *Canadian Conference on Electrical and Computer Engineering 2006*, pages 1716-1721, Ottawa, Canada, May 7-10 2006.

Mohammad Anwar, Truong Nguyen and Marcel Turcotte (2006) Identification of consensus RNA secondary structures using suffix arrays. *BMC Bioinformatics*, 7:244

Truong Nguyen and Marcel Turcotte (2005) Exploring the Space of RNA Secondary Structure Motifs Using Suffix Arrays. *6th International Symposium on Computational Biology and Genome*

Publications (cont.)

Informatics (CBGI 2005). Editors S. Blair et al., Salt Lake City, Utah, USA, July 21-26, 2005, 1291–1298.

Conclusions: Seed

- A suffix tree/array based approach allows us to enumerate a substantial fraction of the search space, using a reasonable amount of resources;
- The search space contains biologically interesting candidates.

Future work

- Adding sequence patterns in the loop regions;
- Developing hybrid algorithms combinatorial + dynamic programming.

Informations

bio.site.uottawa.ca
(home page)
bio.site.uottawa.ca/wiki/space/start
(news)
bio.site.uottawa.ca/software/x-dynalign
(downloads and reprints)
bio.site.uottawa.ca/software/profile-dynalign
(downloads and reprints)
bio.site.uottawa.ca/software/seed
(downloads and reprints)
turcotte@site.uottawa.ca
(E-mail)

Friday's Results

```
$ seed --min_num_stem 3 --max_num_stem 100 --range 2 \
       --save_all_matches --save_as_ct examples/tRNAs-2.

Seed 1.0 [Jul 22 2005] - RNA secondary structure motif i

Copyright (C) 2003-05 University of Ottawa
All Rights Reserved

This program is distributed under the terms of the
GNU General Public License. See the source code
for details.

[ find_all_stems ]
[ size of the motif list is 164 ]
[ filter_by_support ]
[ size of the motif list is 147 ]
```

Friday's Results (contd)

```
[ filter_keep_longest_stems ]
[ size of the motif list is 89 ]
[ fix_all ]
[ size of the motif list is 445 ]
[ combine_all ]
[ generating all 2 stems motifs ]
[ size of the motif list is 445 ]
[ generating all 3 stems motifs ]
[ size of the motif list is 1939 ]
[ generating all 4 stems motifs ]
[ size of the motif list is 1991 ]
[ done ]
[ size of the motif list is 1991 ]
[ postprocess ]
[ size of the motif list is 52 ]
[ elapsed time 1 minutes, 54 seconds ]
```

Friday's Results (contd) (cont.)

```
[ total number of match operations is 520835 ]
[ save_matches_as_ct ]
```

Performance Measures

$A \backslash P$	+	-
+	TP	FN
-	FP	TN

$$\text{Positive Predictive Value (PPV)} = TP/(TP + FP)$$

$$\text{Sensitivity} = TP/(TP + FN)$$

$$\text{Matthews Correlation Coefficient (MCC)} = \sqrt{\frac{\text{TP}}{(\text{TP} + \text{FN})} \times \frac{\text{TP}}{(\text{TP} + \text{FP})}}$$

where A = Actual, P = Predicted, TP = True Positive, FN = False Negative, FP = False Positive and TN = True Negative.

References

Paul P Gardner, Jennifer Daub, John G Tate, Eric P Nawrocki, Diana L Kolbe, Stinus Lindgreen, Adam C Wilkinson, Robert D Finn, Sam Griffiths-Jones, Sean R Eddy, and Alex Bateman. Rfam: updates to the RNA families database. *Nucleic Acids Research*, 37(Database issue):D136–40, Jan 2009.

Pensez-y!

L'impression de ces notes n'est probablement pas nécessaire!