Freie Universität Berlin

MAX-PLANCK-GESELLSCHAFT

RNA folding

W1-High-throughput Genomics, FU Berlin
OWL RNA Bioinformatics, MPI Molgen Berlin
30.11.16

Lernziele

- Einführung von RNA-Molekülen
- Konzept der RNA-Sekundärstruktur
- Lernen wie eine RNA-Sekundärstruktur grafisch darzustellen

- RNA-Strukturvorhersage
  - **Nussinov algorithmus**
    - Die Motivation
    - Wie dynamische Programmierung in diesem Fall angewendet wird
    - Praktisches Beispiel

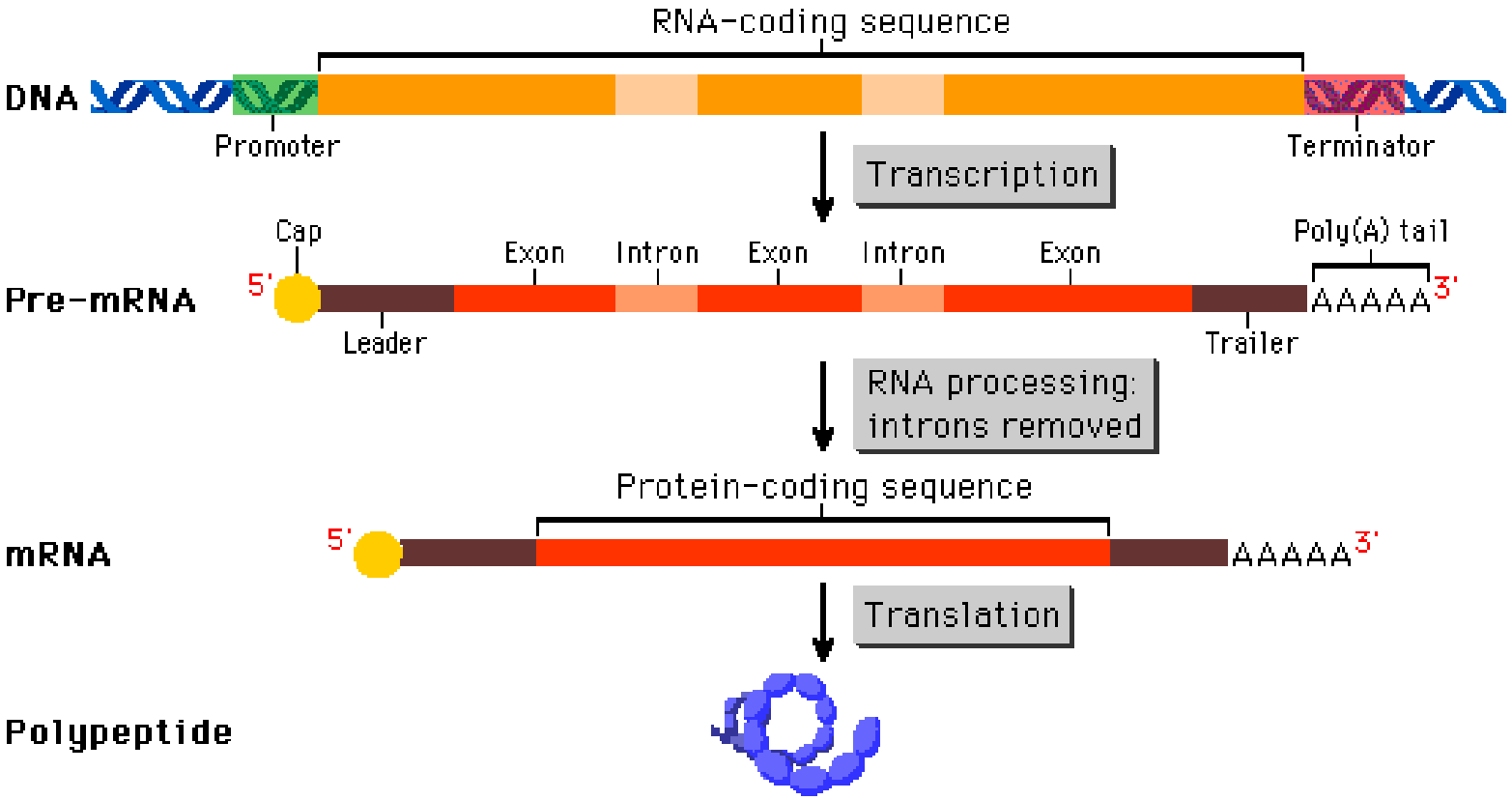
RNA Moleküle

RNA, DNA und Proteine sind die Grundmoleküle des Lebens

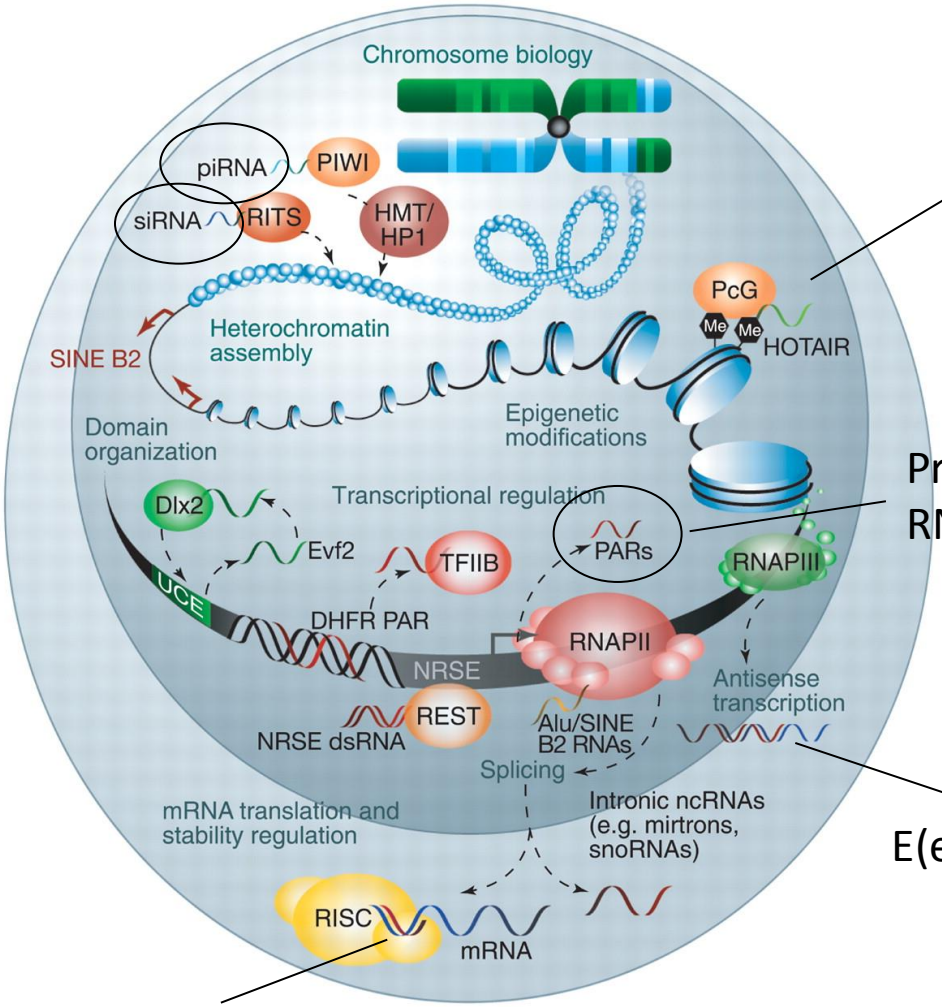
- DNA wird benutzt um die genetische Information zu speichern und replizieren
- Proteine sind die Bausteine der Zelle
- RNAs sind die Zwischenstufen zwischen DNA und Proteinen -> Transkription

Nach der **RNA-Welt-Hypothese** basierte das Leben ursprünglich auf RNA und im Laufe der Zeit delegierten die RNAs das Datenspeicherproblem zur DNA und die katalytische Funktionalität zu Proteinen.

Beispiel: Der Prozess der Transkription



Die wachsende Bedeutung der RNAs: die RNA-Welt



lncRNAs

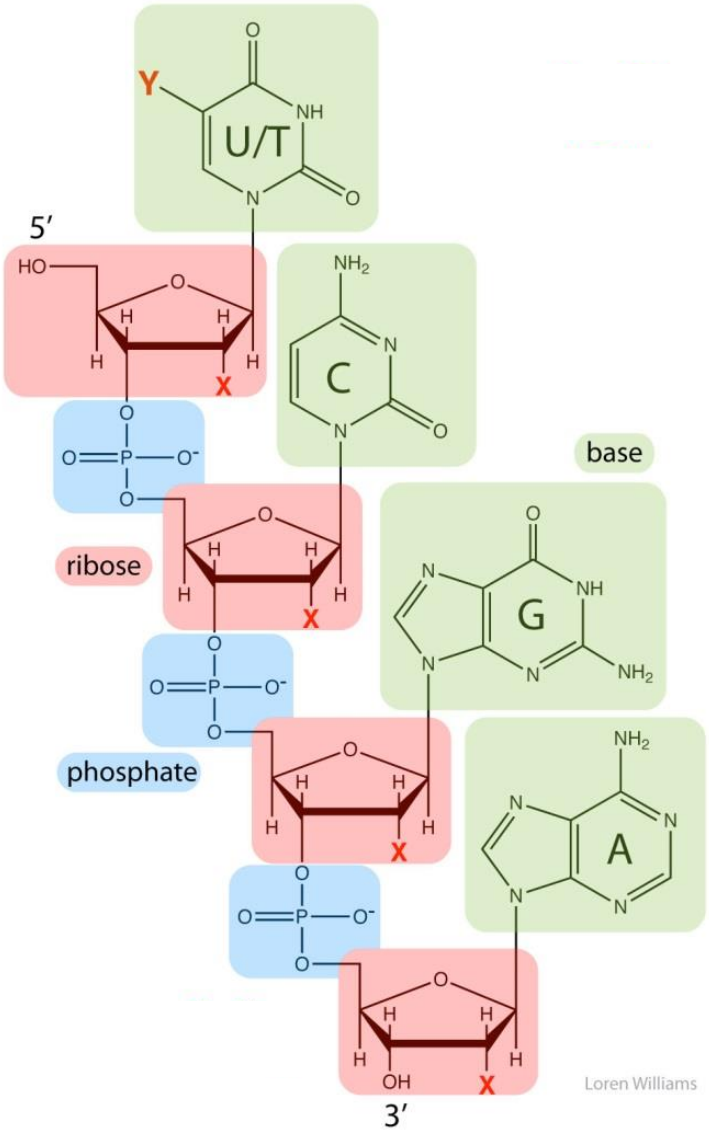
Promoter-associated RNAs

E(enhancer-like)RNAs

miRNAs

Amaral et al. Science 2008;319:1787-1789

RNA Rückgrat



Ein RNA-Molekül ist ein Polymer aus vier Arten von Ribonukleotiden, jeweils durch eine der vier Basen angegeben.

A -> adenine

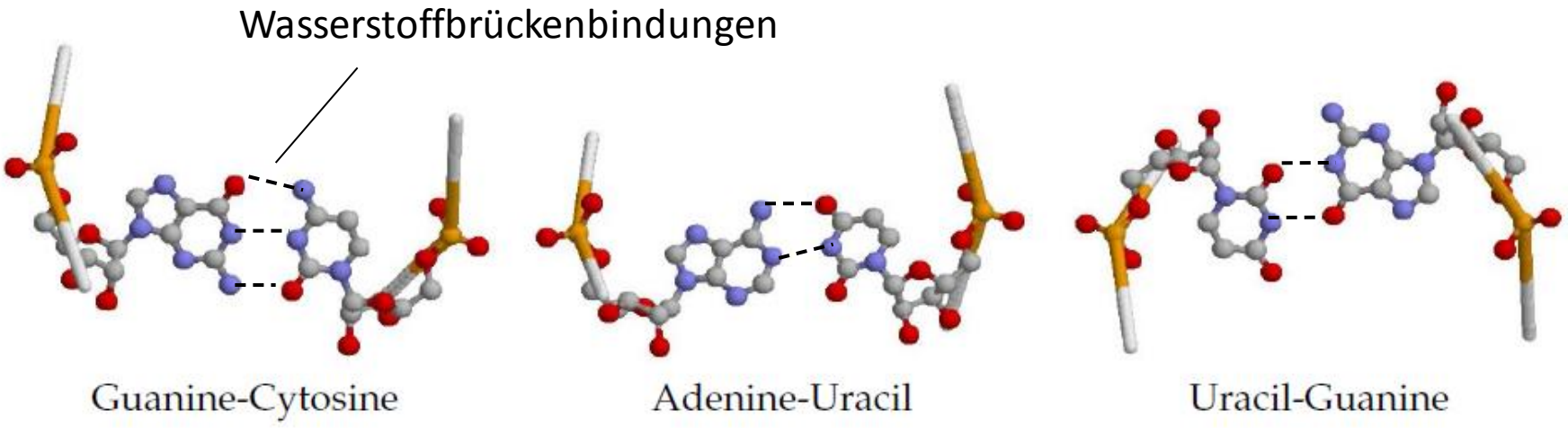
C -> cytosine

G -> guanine

U(T) -> uracil

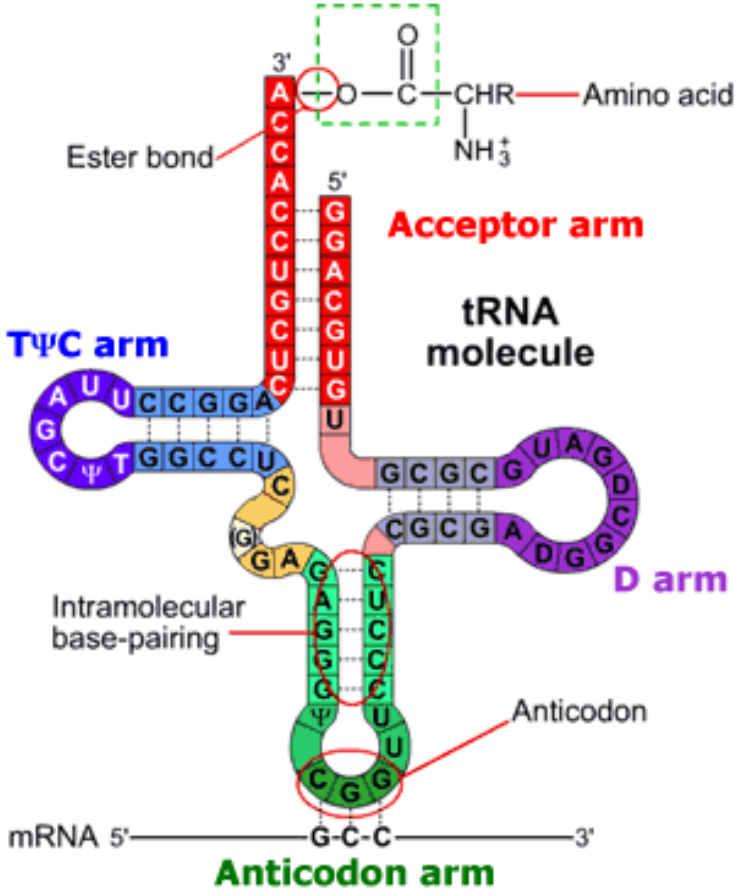
RNA-Sekundärstruktur

Im Gegensatz zu DNA, ist RNA einzelsträngig. Allerdings formen die komplementären Basen C-G und A-U stabile Basenpaare über Wasserstoffbrückenbindungen. Diese werden Watson-Crick-Paare genannt. Wichtig sind auch die schwächeren U-G Wobble-Paare. Zusammen werden sie „kanonische Basenpaare“ genannt.



RNA-Sekundärstruktur

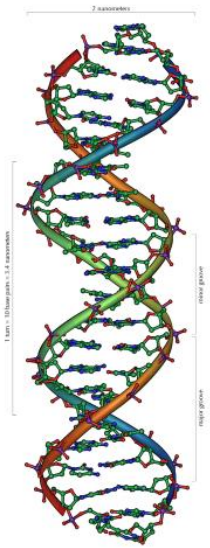
Die Basenpaare, die zwischen den verschiedenen Teilen eines RNA-Molekül gebildet werden, definieren die Sekundärstruktur des RNA-Moleküls.
Hier ist die Sekundärstruktur von einem tRNA:



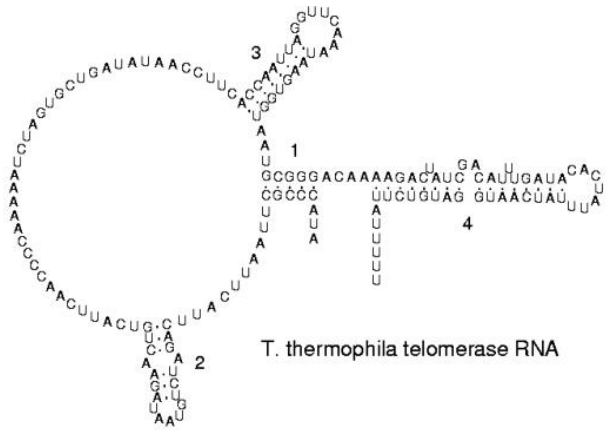
Sekundärstruktur: Satz von Basenpaaren, die auf einer Ebene abgebildet werden können

Zusammenfassen: DNA vs RNA

1. DNA doppelsträngig, **RNA einzelsträngig**, oft Sekundaerstruktur durch intramolekulare Wasserstoffbrucken
2. **Länge**: DNA Milionen von Basenpaaren, RNA ~20 bis mehrere tausend Nukleotide
3. Biochemie: RNA Ribose/ Uracil (U), DNA 2-Desoxyribose / Thymin (T)
4. Biologischen Rollen: DNA Trägerin der Erbinformation, RNA unterschiedliche Funktionen. Im Gegensatz zur DNA spielt die Struktur der RNA bei deren Funktion eine wesentliche Rolle.



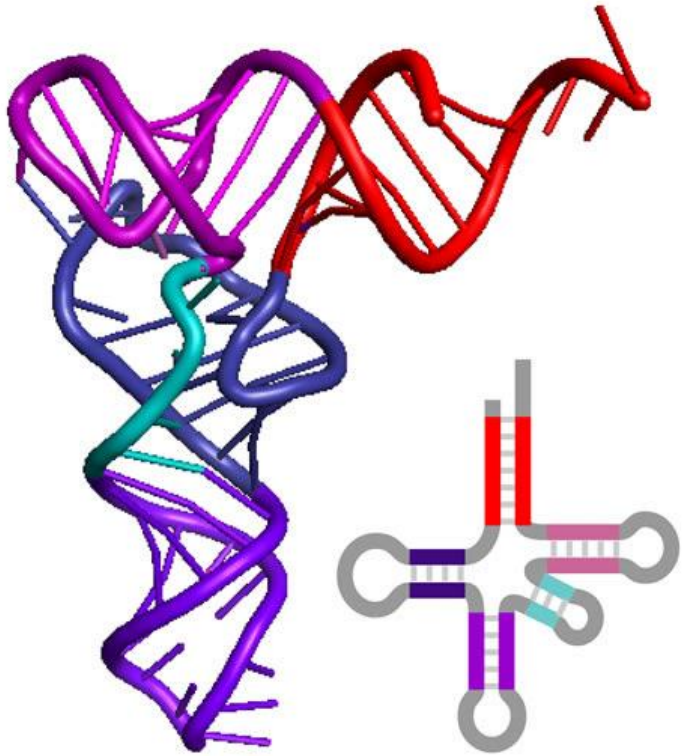
DNA



RNA

Struktur Konformationen von RNA

- Primärstruktur: Sequenz von Monomeren ATGCCGTCAC..
- Secondärstruktur: 2D-Faltung, durch Wasserstoffbrückenbindungen definiert
- Tertiärstruktur: 3D-Faltung
- Quartärstruktur: komplexe Anordnung von mehreren gefalteten Moleküle



RNA-Sequenzen und Strukturen

- Die RNA Sequenz faltet sich selbst zurück aufgrund der Komplementarität der Basen.
- RNA-Strukturbestimmung ist experimentell schwierig, **daher ein wichtiges Thema fur die Bioinformatik**.

- Die Methoden zur 2D-Vorhersage konnen in zwei Gruppen eingeteilt werden:
  - Methoden, die ihre Vorhersagen aus MSA ableiten
  - Methoden, die ihre Vorhersagen für **einzelne Sequenzen bestimmen** (Maximierung Anzahl Basenpaare oder Minimierung freie Energie)

Formale Definition der RNA-Struktur

Die echte Sekundärstruktur eines RNA-Moleküls ist die Menge der Basenpaare im dreidimensionale Raum.

Definition Für unsere Zwecke ist ein RNA-Molekül einfach ein String

$$x = (x_1.x_2, \ldots . x_L)$$
mit $x_i \, \epsilon \{A, C, G, U\}$ fur alle i

Definition Eine Sekundärstruktur für x ist eine Menge P von geordneten Basenpaaren *(i,j),* mit $1 \leq i \leq j \leq L$ und:

1. **Primary proximity Constraint**:
   $|i\text{-}j| > 3$, d.h. die Basenpaaren dürfen nicht zu nahe beieinander liegen

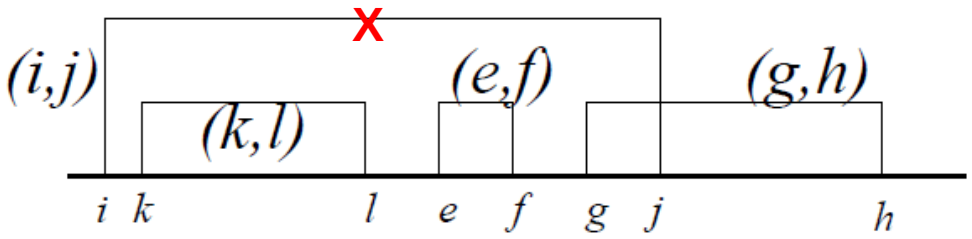
2. **Nesting Constraint**:
   Sind (i,j) und (i‘,j‘) zwei Paare von Nukleotiden, wobei i < i‘ < j, dann gilt j < j‘.
   d.h. die Basenpaare dürfen sich nicht überschneiden.

Formale Definition der RNA-Struktur

Verschachtelte Strukturen

Im Folgenden werden wir nur verschachtelten Sekundärstrukturen behandeln, da komplizierteren nicht-verschachtelten Strukturen nicht mehr mit unsere Methoden lenkbar sind.

Diese ”Schachtelungsbedingung” verbietet überkreuzte Wasserstoffbrücken, erlaubt dagegen geschachtelte Wasserstoffbrücken. Überkreuzte Wasserstoffbrücken, so genannte Pseudoknoten, kommen relativ selten vor. Algorithmen, welche Pseudoknoten zulassen, sind wesentlich weniger effizient als solche, die sie verbieten.



Strukturvorhersage durch Maximierung Von Basenparungen

- Lass uns mit diese Sequenz spielen: **UUGACAUCG**
- Ziel:die Sekundärstruktur mit der maximalen Anzahl an Basenpaaren finden, wobei zwischen zwei paarenden Basen mindestens eine Ungepaarte stehen soll

erlaubt

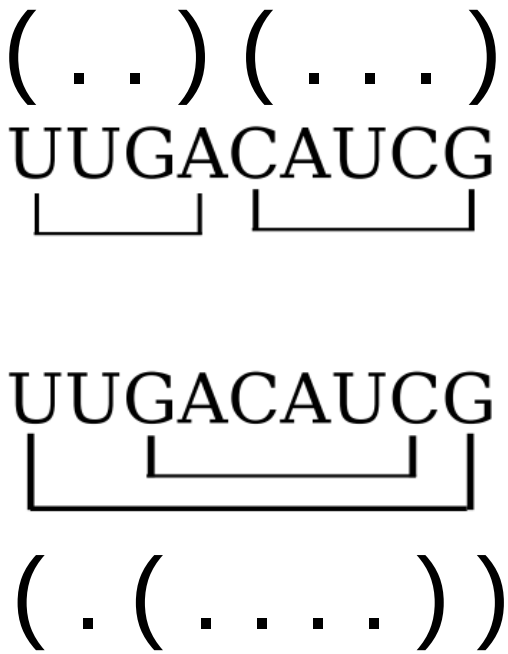
erlaubt

nicht erlaubt

nicht erlaubt

RNA-Struktur: Klammern und Punkte

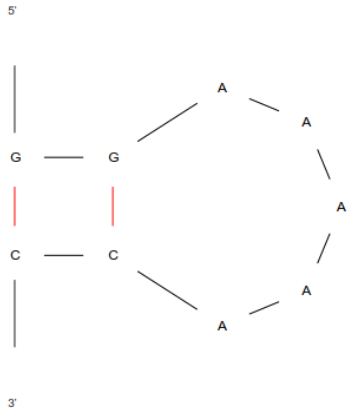
Wir können die RNA-Struktur als Strings mit balanzierten Klammern und Punkten mit der entsprechenden Schachtelebene (nesting level) darstellen



RNA-Struktur: Klammern und Punkte

- Die Sekundärstruktur kann von der Sequenz und Klammerndarstellung ermittelt werden
- Beispiel:

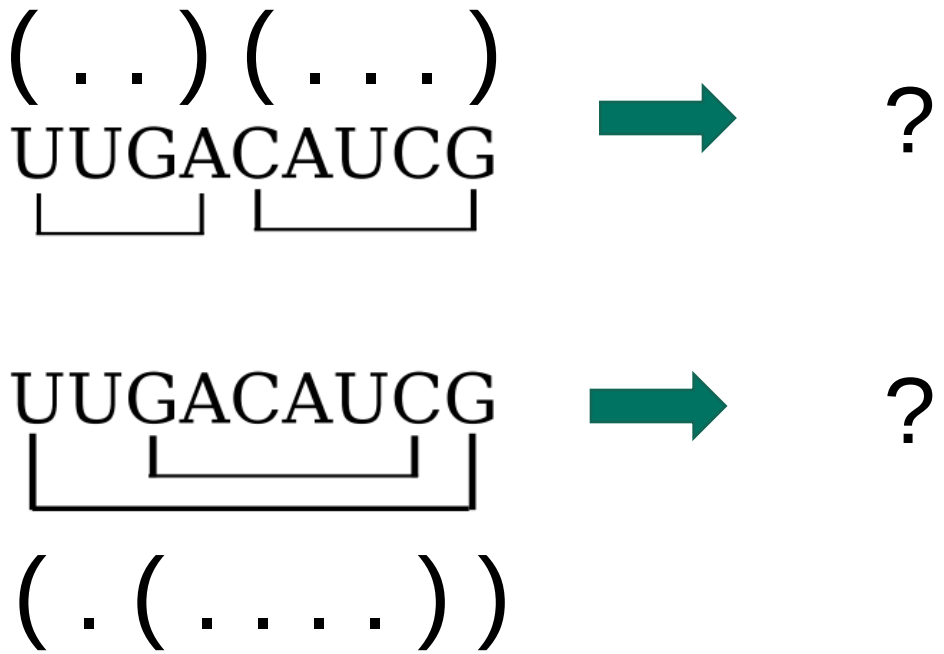
```
GGAAAAACC
((.....))
```



Graph Darstellung

RNA-Struktur: Klammern und Punkte

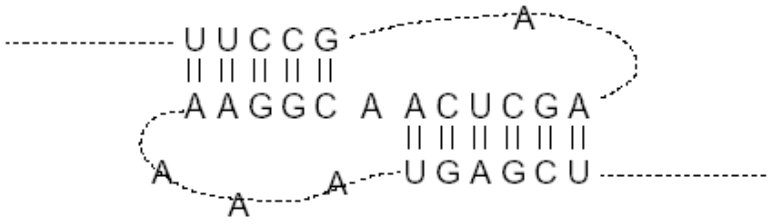
- Die Sekundärstruktur kann von der Sequenz und Klammerndarstellung ermittelt werden

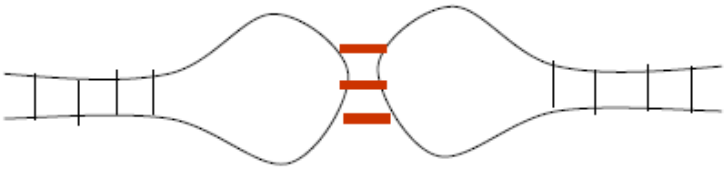


Nicht-verschachtelte Interaktionen

Interaktionen, die nicht verschachtelt sind, führen zu einer ‚Pseudoknoten'-Struktur oder ‚kissing hairpins': Segmente der Sequenz, die in die gleiche Richtung verbunden sind oder dreidimensionale Kontakte haben:

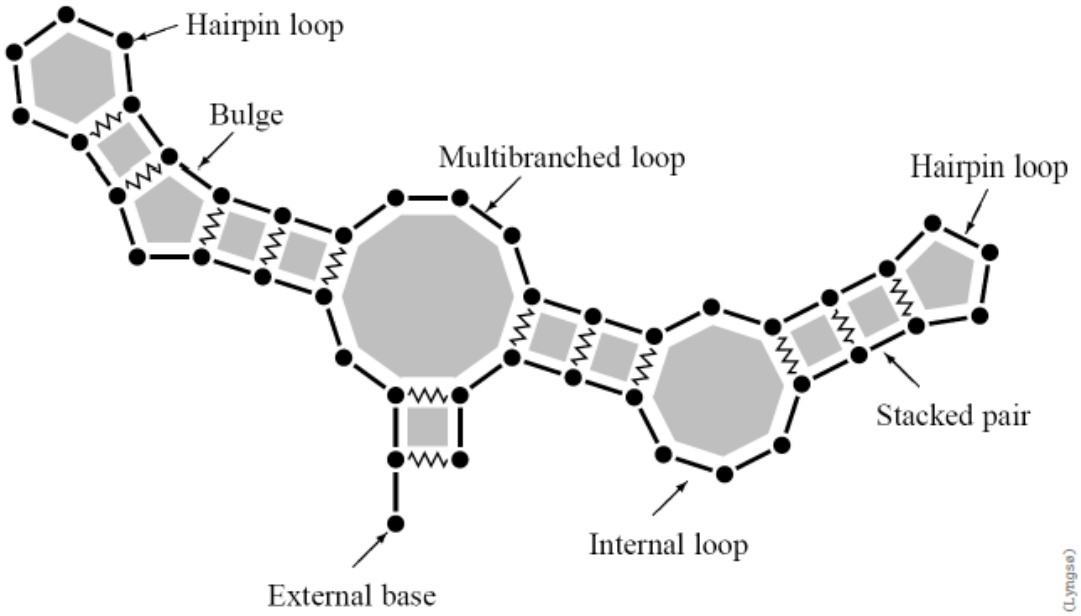
Pseudoknot





Kissing hairpins

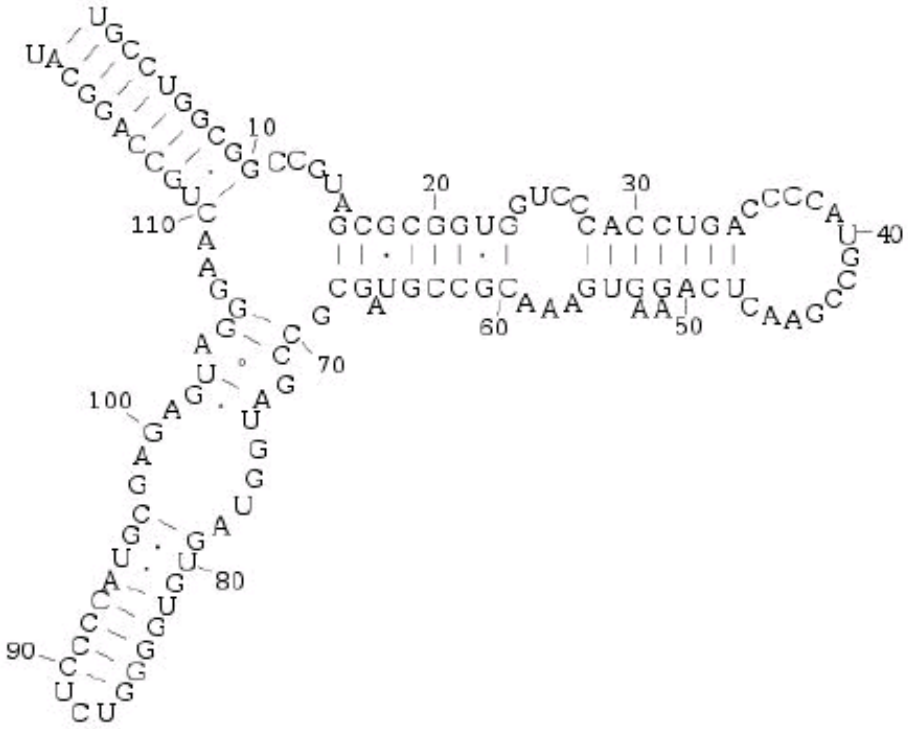
RNA-Sekundärstrukturelemente



- single stranded RNA
- stacked base pairs
- stem & loop (hairpin loop)
- bulge loop
- interior loop
- junction or multi-loop

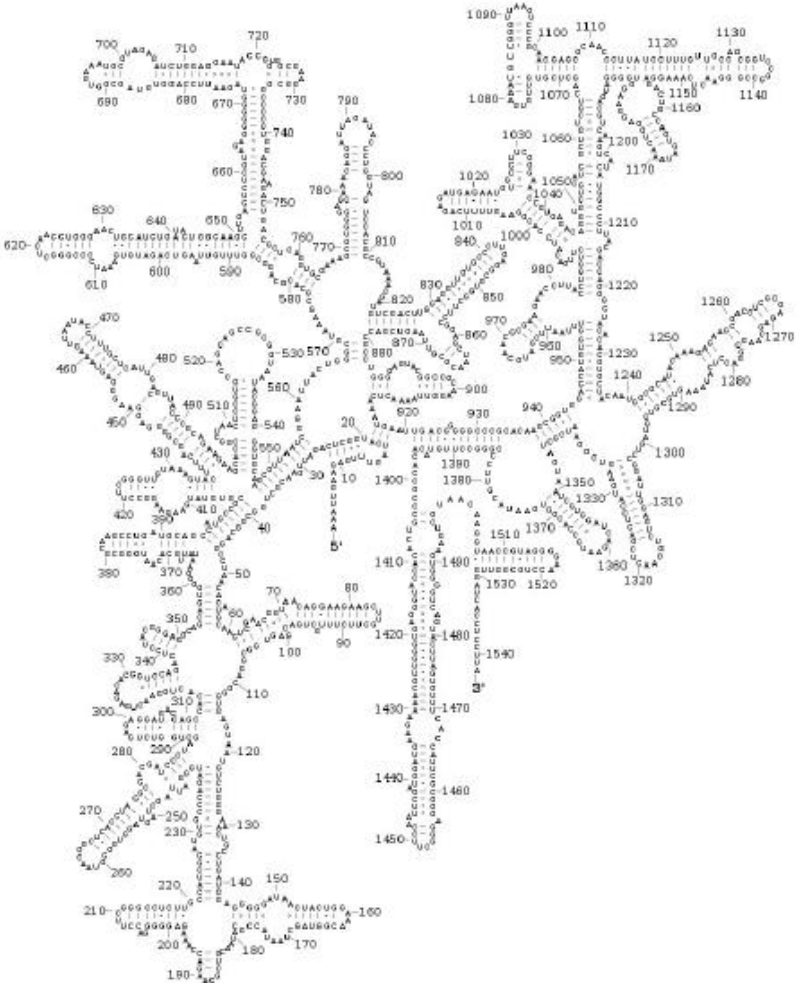
Typen von Einzel- und Doppelstrang-Regionen
Dies nennt man Basenpaar **Graphdarstellung**

RNA-Struktur Beispiel 1: 5s rRNA



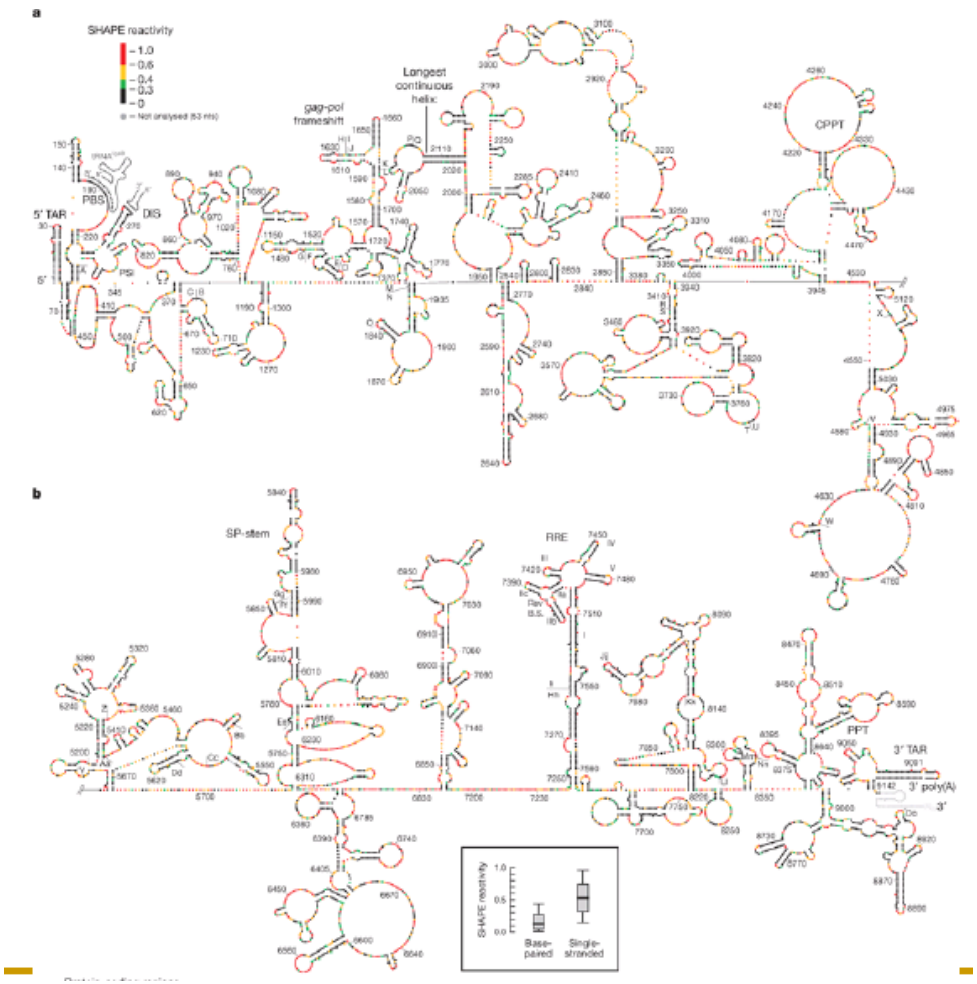
E. coli 5S
120 bases

RNA-Struktur Beispiel 2 : E.coli 16S rRNA



1542 bases

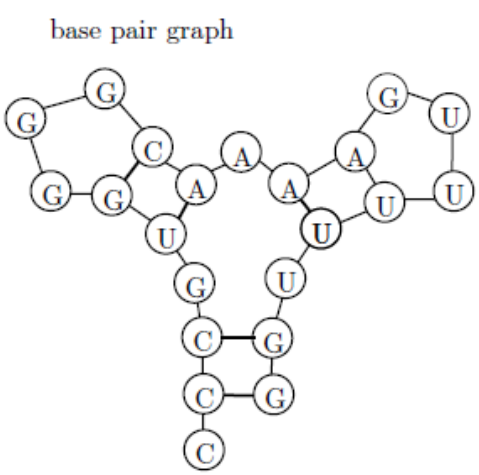
RNA-Struktur Beispiel 1 : HIV

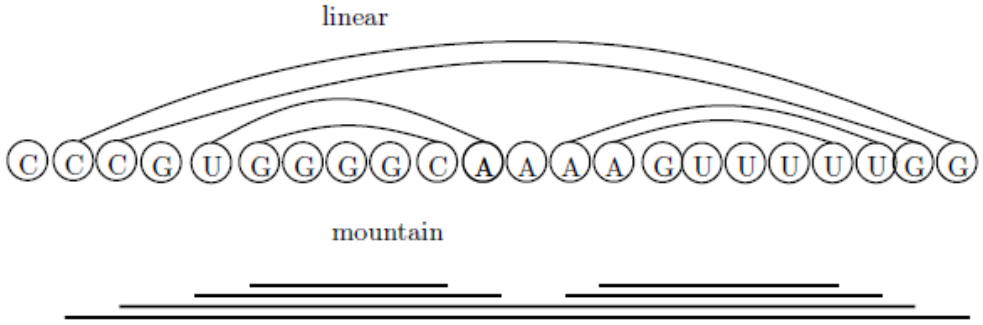


9173 basis

Watts et al., Nature 2013

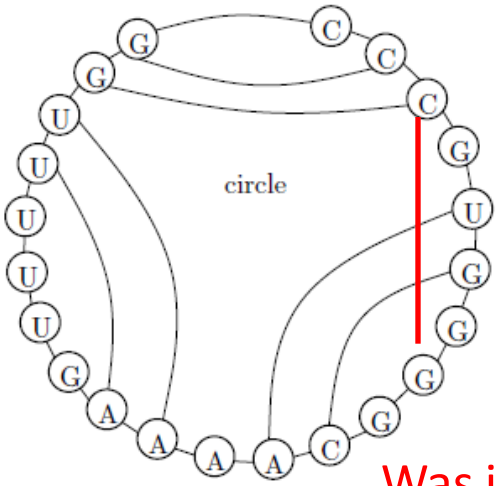
Darstellungstypen einer RNA-Sekundärstruktur

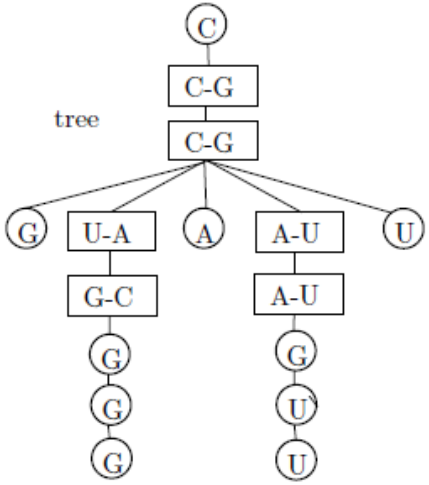




bracket

((((***))*((***))*))





Was ist das?

Vorhersage der RNA-Sekundärstruktur

Ansatz:
Finde eine Konfiguration, die die **Anzahl von Basenpaaren maximiert**
Für eine Sequenz der Länge L, wächst die Anzahl der möglichen Konfigurationen exponentiell mit der Länge der Sequenz.
Dabei ist es unmöglich, alle möglichen Strukturen zu aufzählen!

Zum Glück, können wir ***Dynamische Programmierung*** verwenden, um die effizienteste Lösung zu finden.
Eine Methode , um das zu tun, hat Ruth Nussinov in 1978 publiziert.

Der Algorithmus ist rekursiv. Es berechnet die beste Struktur für kleinere Teilfolgen größere Sequenzen. Von der kleinsten bis zur vollständigen Sequenz.

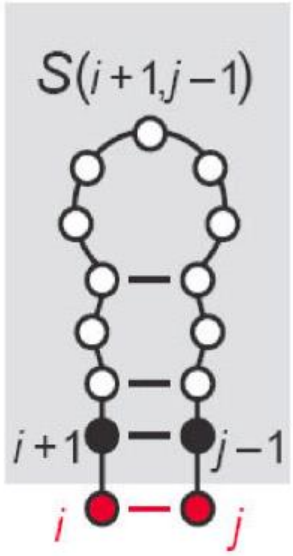
Nussinov Faltungalgorithmus -I (1978)

- Eine vereinfachte Version des Nussinov-Algorithmus versucht, die Anzahl der gepaarten Basen zu maximieren
- Unser Score: +1 für Basenpaar, 0 für alles Andere
- Sei $x_1 \ldots . x_L$ ist eine Sequenz von 1 zu L faltenden Nukleotiden.
- Berechne die maximale Anzahl von gebildeten Basenpaaren der Teilfolge $x[i:j]$
- $\boldsymbol{S(i,j)}$ Max score für die Subsequenz $i, i+1, \ldots . j-1, j$
- $\boldsymbol{S(i,j)}$ kann rekursiv berechnet werden (**Dynamic Programming**)
- Und die Struktur rekursiv gefaltet werden
  (gegeben das wir bereits für alle kurzen Sequenzen $x[m:l]$ $i < m < l < j$
  berechnet haben)

Nussinov Faltungalgorithmus -I (1978)

Die Struktur auf $x[i:j]$, kann auf vier Arten berechnet werden:

1) Falls *i* , *j* ein WC-Basenpaar sind



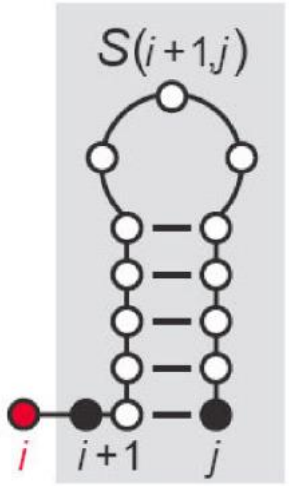
Füge die gepaarten Basen i und j der besten Struktur für Subsequenz [i+1,j-1] hinzu

$$S(i,j) = 1 + S(i+1, j+1)$$

Nussinov Faltungalgorithmus -I (1978)

Die Struktur auf $x[i:j]$, kann auf vier Arten berechnet werden:

2) Falls *i* ungepaart bleibt



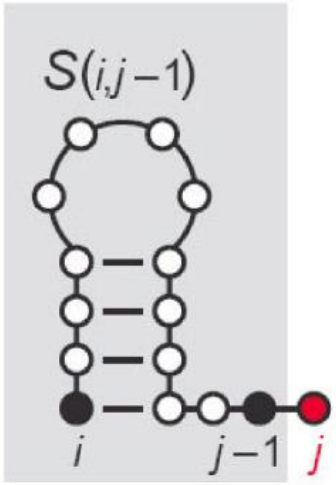
Füge die ungepaarten Base i der besten Struktur für Subsequenz [i+1,j] hinzu

$$S(i,j) = S(i+1,j)$$

Nussinov Faltungalgorithmus -I (1978)

Die Struktur auf $x[i:j]$, kann auf vier Arten berechnet werden:

3) Falls *j* ungepaart bleibt



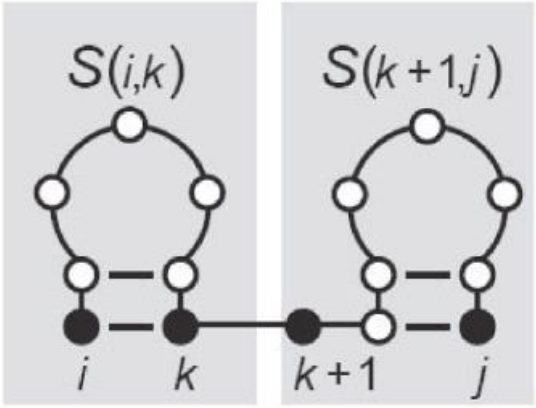
Füge die ungepaarten Base j der besten Struktur für Subsequenz [i,j-1] hinzu

$$S(i,j) = S(i,j-1)$$

Nussinov Faltungalgorithmus -I (1978)

Die Struktur auf $x[i:j]$, kann auf vier Arten berechnet werden:

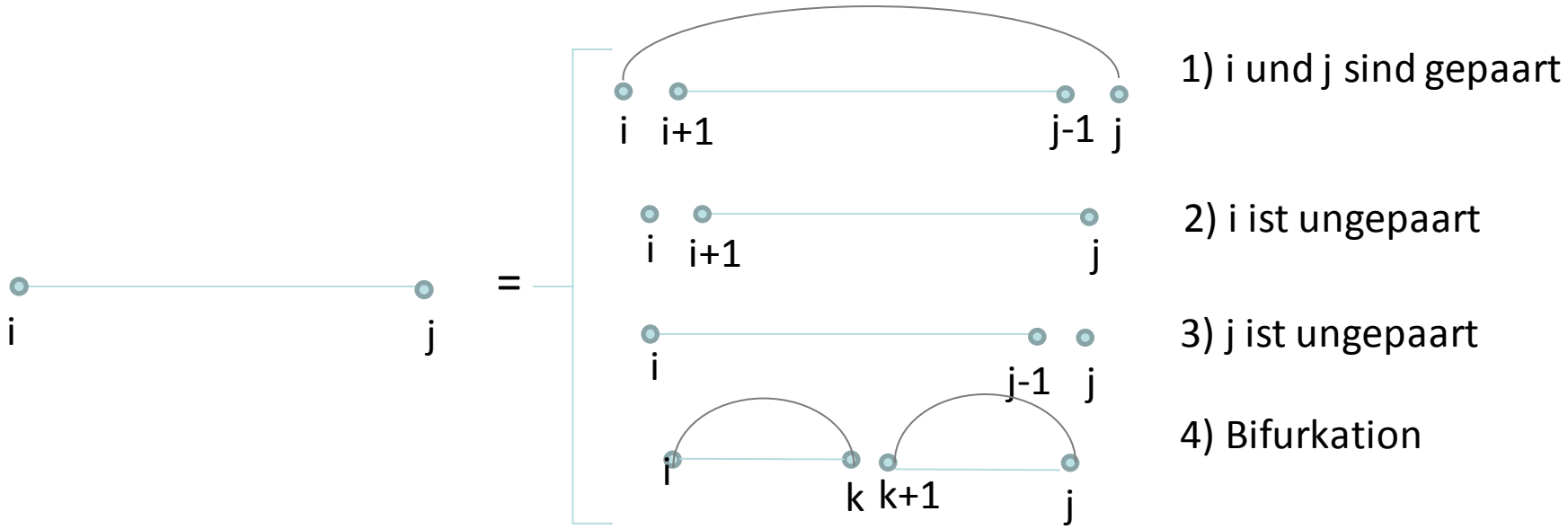
4) Falls i , j jeweils mit anderen Nukleotiden gepaart sind, handelt es sich um eine Bifurkation



Struktur S(i , j) besteht dann aus den Strukturen für zwei Subsequenzen **i , . . . , k** und **k + 1, . . . , j** .
(Kombiniere zwei optimale Strukturen [i,k], [k+1,j])

$$S(i,j) = max_{i<k<j}(S(i,k) + S(k+1,j))$$

Graphische Darstellung des Faltungsalgorithmus



1) i und j sind gepaart

2) i ist ungepaart

3) j ist ungepaart

4) Bifurkation

Nussinov Faltungalgorithmus -I (1978)

Der Algorithmus der dynamischen Programmierung hat zwei Phasen:

- Der ***fill*** Schritt**,** berechnet rekursiv die besten Scores $S(i,j)$,
  welche die maximale Anzahl an bp darstellen,
  die in einer Subsequenz $x_i \ldots . x_j$ gefunden werden

- Der ***traceback*** Schritt, traceback durch die berechnete Matrix,
  um die beste Struktur mit der maximale Anzahl an bp zu finden

Nussinov Algorithmus: Fill Phase

Algorithm (Nussinov RNA faltung, fill schritt) – Beispiel, Sequenz: GGGAAAUCC

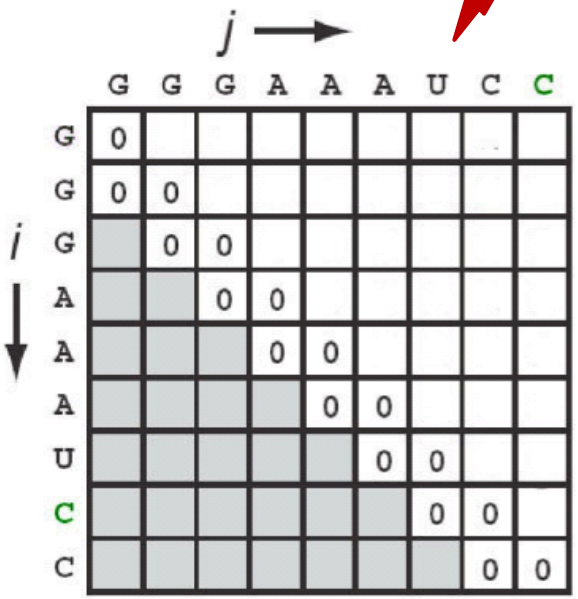
Input: Sequenz $x = (x_1, x_2, \ldots \ldots x_L)$

Output: Maximale Anzahl S_{max} von Basenpaare für Sequenz *x*

Initialisierung der DP Matrix:

```
for i=2 to L do
        S[i,i-1]=0
for i=1 to L do
        S[i,i]=0
```

Nur die Haelfte der Matrix wird verwendet. Warum?



Nussinov Algorithmus: Fill Phase

Algorithm (Nussinov RNA folding, fill stage) – Beispiel, Sequenz: GGGAAAUCC

Input: Sequenz $x = (x_1, x_2, \dots \dots x_L)$
Output: Maximale Anzahl S_{max} von Basenpaare für Sequenz *x*

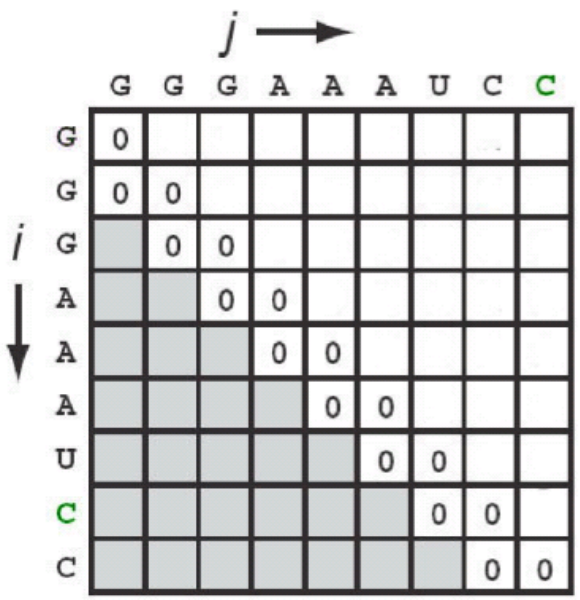
Initialisierung der DP Matrix:

```
for i=2 to L do
       S[i,i-1]=0
for i=1 to L do
       S[i,i]=0
```

Rekursion (Fullung der DP Matrix):

```
for n=2 to L do
  for j=n to L do
    i=j-n+1
```

$$S = \max \begin{cases} S(i+1, j) \\ S(i, j-1) \\ S(i+1, j-1) + \delta(i, j) \\ \max_{i<k<j}(S(i,k) + S(k+1, j)) \end{cases}$$



Die DP Rekursion pruft vier Moeglichkeiten (1,2,3,4 von der vorherigen Folie)
$\delta(i, j)$=1 for base pairs, $\delta(i, j)$=0 otherwise

Nussinov Algorithmus: Fill Phase

Algorithm (Nussinov RNA folding, fill stage) – Beispiel, Sequenz: GGGAAAUCC

Input: Sequenz $x = (x_1, x_2, \dots \dots x_L)$
Output: Maximale Anzahl S_{max} von Basenpaare für Sequenz *x*

Initialisierung der DP Matrix:

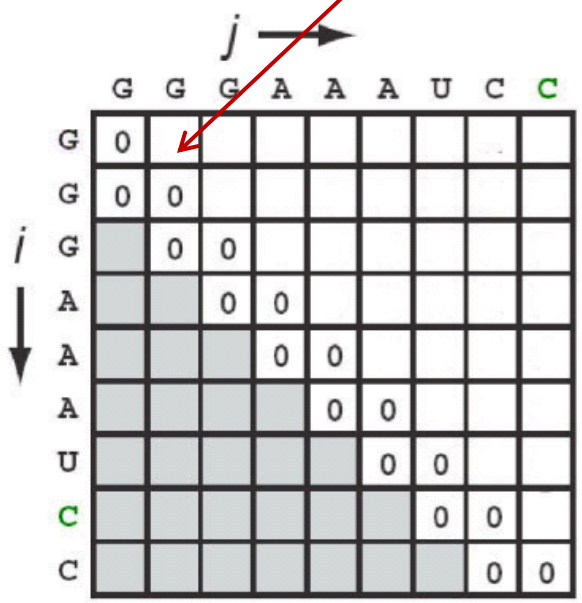
```
for i=2 to L do
        S[i,i-1]=0
for i=1 to L do
        S[i,i]=0
```

Rekursion (Fullung der DP Matrix):

```
For n=2 to L do
  for j=n to L do
    i=j-n+1
```

$$S=\max\begin{cases} S(i+1,j) \\ S(i,j-1) \\ S(i+1,j-1)+\delta(i,j) \\ \max_{i<k<j}(S(i,k)+S(k+1,j)) \end{cases}$$

Welcher Wert ist hier?

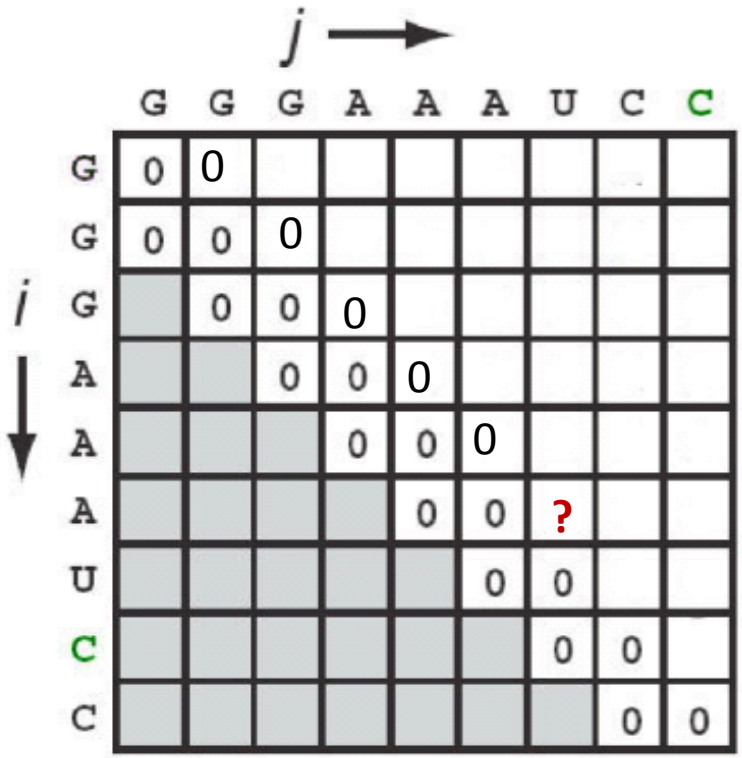


Nussinov Algorithmus: Fill Phase

Rekursion (Fullung der DP Matrix):

```
For n=2 to L do
  for j=n to L do
    i=j-n+1
```

$$S = \max \begin{cases} S(i+1, j) \\ S(i, j-1) \\ S(i+1, j-1) + \delta(i, j) \\ \max_{i<k<j}(S(i, k) + S(k+1, j)) \end{cases}$$

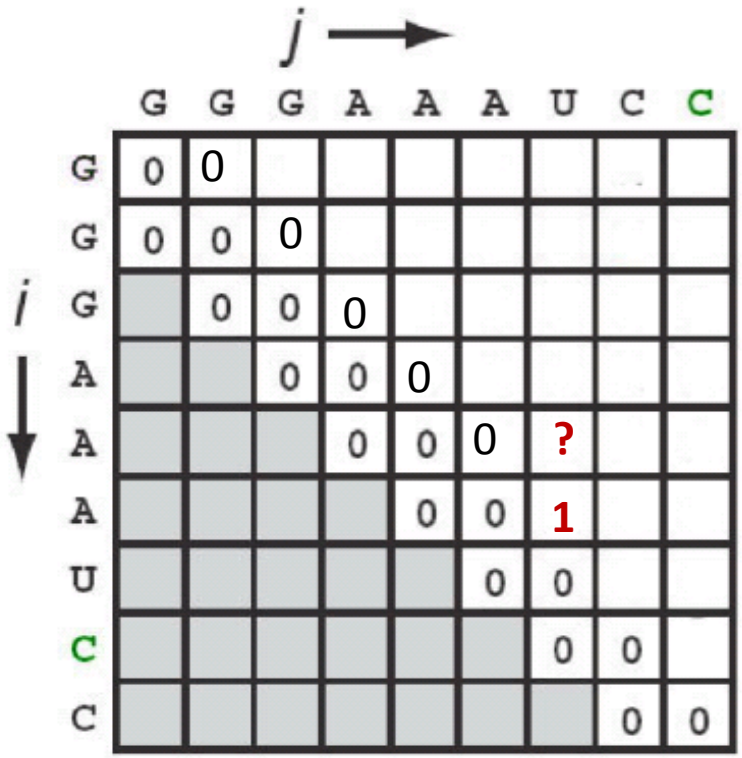


Nussinov Algorithmus: Fill Phase

Rekursion (Fullung der DP Matrix):

```
For n=2 to L do
  for j=n to L do
    i=j-n+1
```

$$S=\max\begin{cases} S(i+1,j) \\ S(i,j-1) \\ S(i+1,j-1)+\delta(i,j) \\ \max_{i<k<j}(S(i,k)+S(k+1,j)) \end{cases}$$

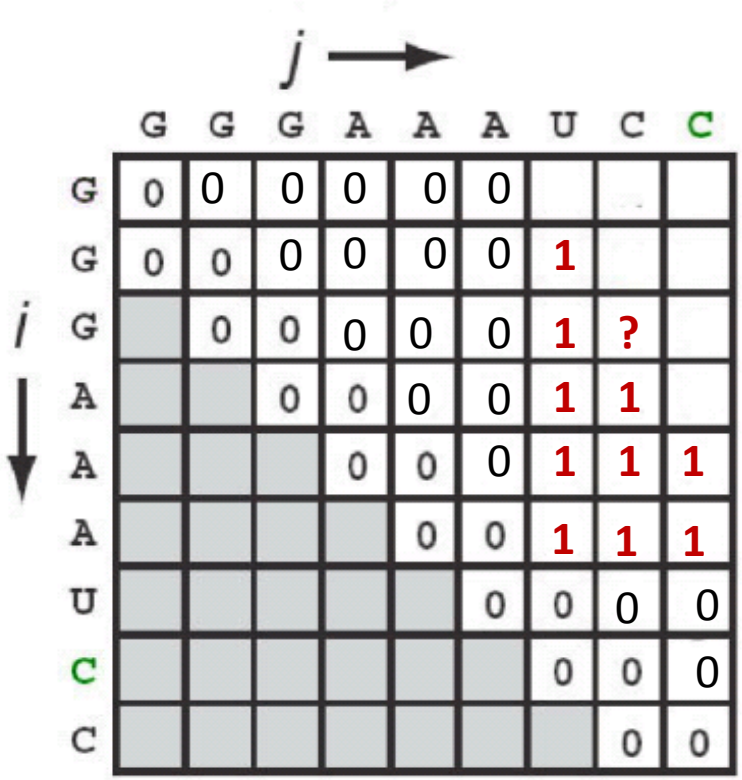


Nussinov Algorithmus: Fill Phase

Rekursion (Fullung der DP Matrix):

```
For n=2 to L do
  for j=n to L do
    i=j-n+1
```

$$S=\max\begin{cases} S(i+1,j) \\ S(i,j-1) \\ S(i+1,j-1)+\delta(i,j) \\ \max_{i<k<j}(S(i,k)+S(k+1,j)) \end{cases}$$

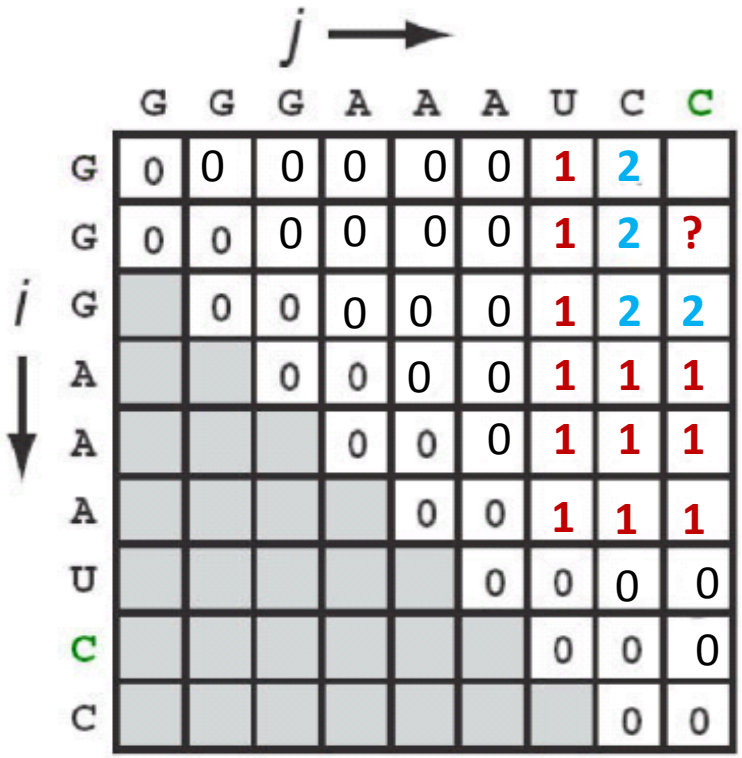


Nussinov Algorithmus: Fill Phase

Rekursion (Fullung der DP Matrix):

```
For n=2 to L do
  for j=n to L do
    i=j-n+1
```

$$S = \max \begin{cases} S(i+1, j) \\ S(i, j-1) \\ S(i+1, j-1) + \delta(i, j) \\ \max_{i<k<j}(S(i, k) + S(k+1, j)) \end{cases}$$

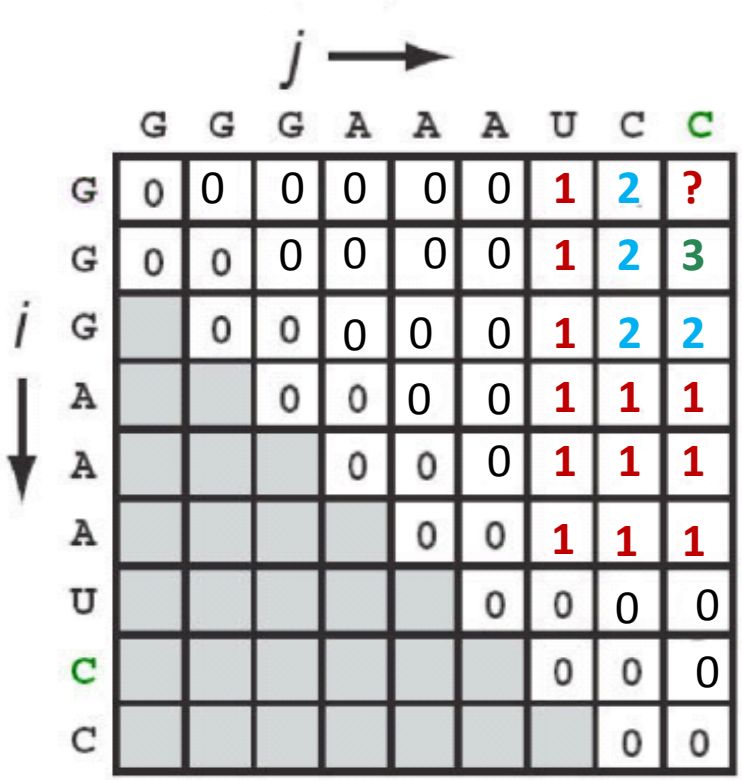


Nussinov Algorithmus: Fill Phase

Rekursion (Fullung der DP Matrix):

```
For n=2 to L do
  for j=n to L do
    i=j-n+1
```

$$S=\max\begin{cases} S(i+1,j) \\ S(i,j-1) \\ S(i+1,j-1)+\delta(i,j) \\ \max_{i<k<j}(S(i,k)+S(k+1,j)) \end{cases}$$

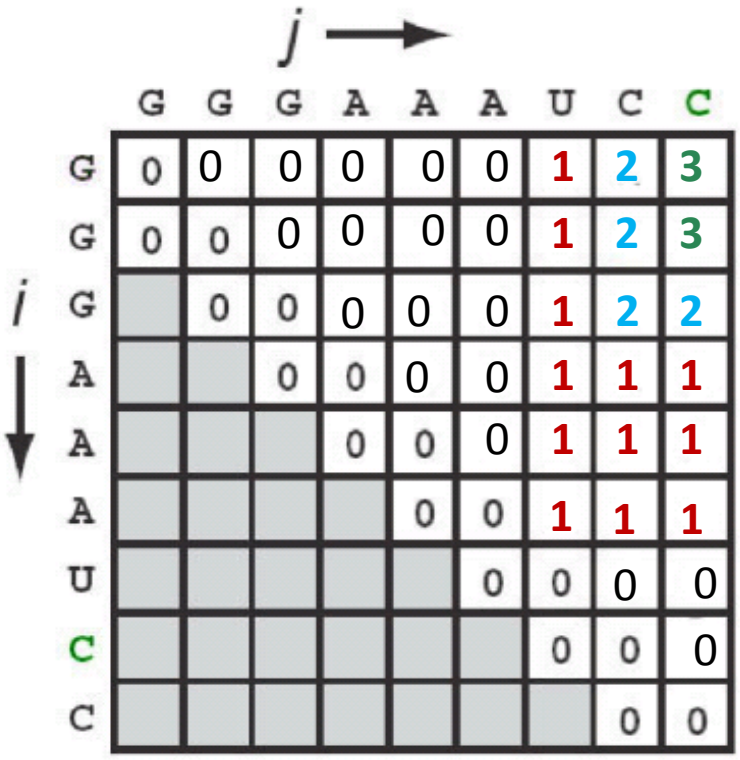


Nussinov Algorithmus: Fill Phase

Recursion (Fullung der DP Matrix):

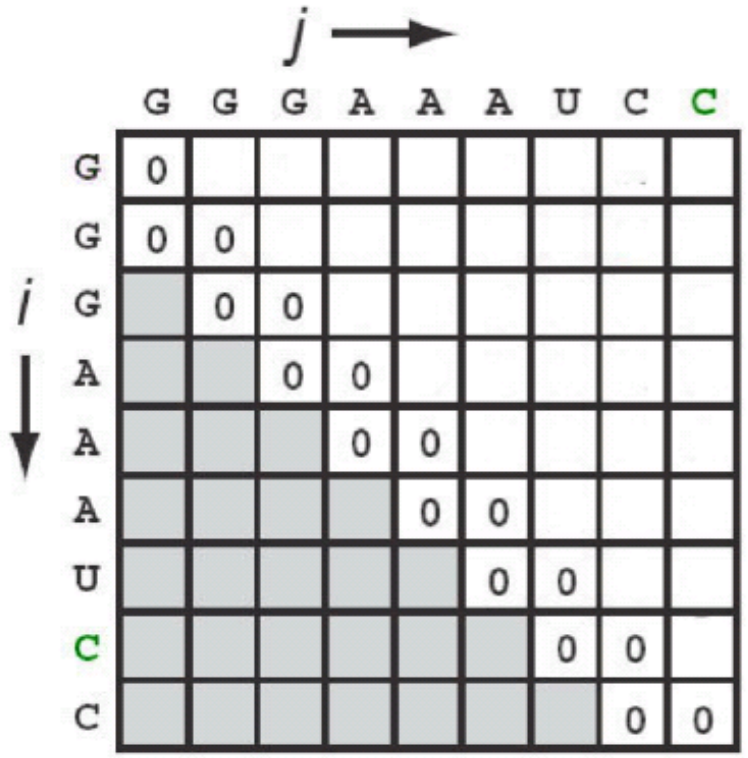
```
For n=2 to L do
  for j=n to L do
    i=j-n+1
```

$$S=\max\begin{cases} S(i+1,j) \\ S(i,j-1) \\ S(i+1,j-1)+\delta(i,j) \\ \max_{i<k<j}(S(i,k)+S(k+1,j)) \end{cases}$$



Die letzte Zell (1,9) gibt den Score einer optimalen Struktur für die ganze Sequenz. Die Optimale Score kann hier durch Option 1 und 2 erreicht werden. **Warum?**

Nussinov Algorithmus: Fill Phase



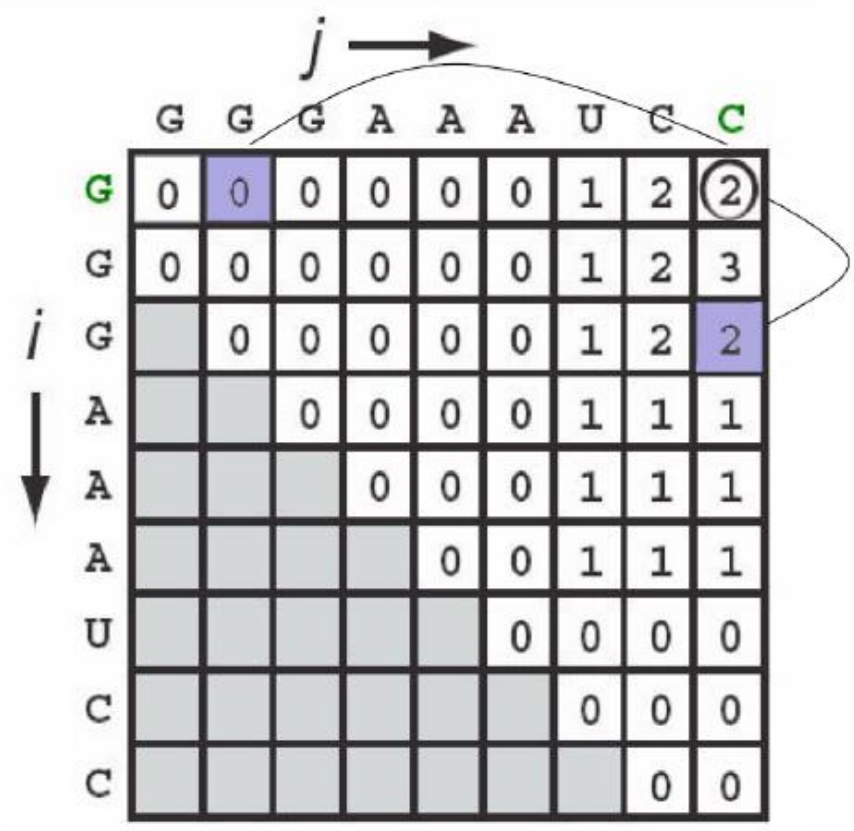
Achtung! Wir müssen immer noch den Score für Bifurkationen berechnen: k=2,3,4,5,6,7,8

Nussinov Algorithmus: Fill Phase

k=2:
Wir haben zwei Subsequenzen
(i,k) und (k+1,j)
(1,2) und (3,9)
Der Score ist S(1,2)+S(3,9) =2

Es geht weiter für k=3,4,5,6,7,8

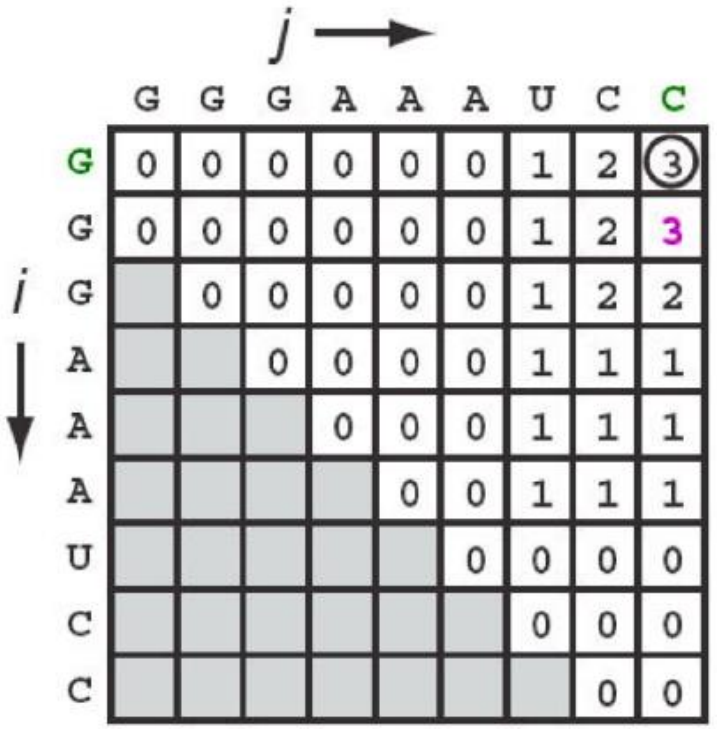
Maximalwert ist 2
erreicht wenn k=2
0+2=2



Nussinov Algorithmus: Fill Phase

S(1,9) = max{2,3,2,2}=3

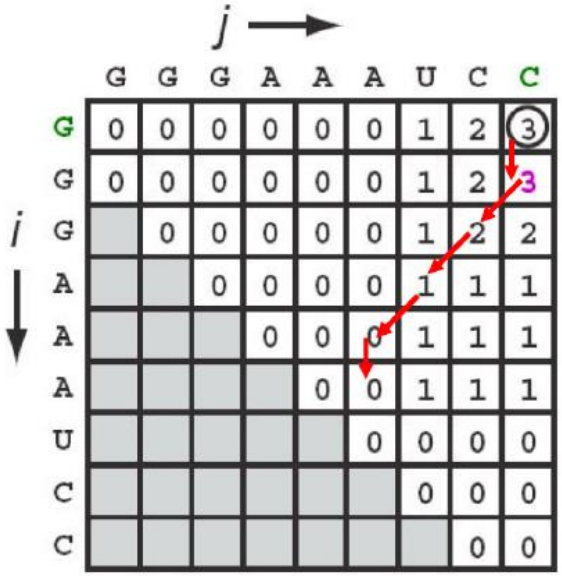
Traceback, um die Struktur selbst zu finden

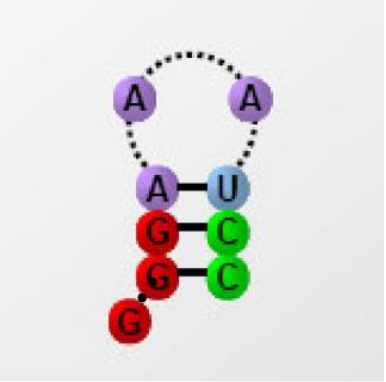


Traceback in die Nussinov Matrix

Bauen die RNA Struktur

- Der Traceback beginnt immer in der rechten oberen Ecke der Matrix und verfolgt den Pfad einer optimalen Struktur
- In diesem Fall ist der Traceback relativ einfach, weil keine verzweigte RNA-Strutkur vorliegt
- man beginnt mit Zelle (1, L) die Scores der drei umgebenden zellen (links, schräg links unten und unten)
- vergleichen und so schlussfolgern, wie der Score in der aktuellen Zelle zustande gekommen sein muss und so eine optimale Struktur finden.
- Oft (wie hier) sind viele Pfade mit demselben Score möglich





Nussinov - Der Traceback Schritt

Algorithm traceback(i, j) (Nussinov RNA folding)
Input: Matrix γ and positions i, j.
Output: Secondary structure maximizing the number of base pairs.
Initial call: traceback$(1, L)$.

```
if i < j then
    if γ(i, j) = γ(i + 1, j) then                          // case (1)
        traceback(i + 1, j)
    else if γ(i, j) = γ(i, j − 1) then                     // case (2)
        traceback(i, j − 1)
    else if γ(i, j) = γ(i + 1, j − 1) + δ(i, j) then       // case (3)
        print base pair (i, j)
        traceback(i + 1, j − 1)
    else for k = i + 1 to j − 1 do                         // case (4)
        if γ(i, j) = γ(i, k) + γ(k + 1, j) then
            traceback(i, k)
            traceback(k + 1, j)
```

Komplexität des Nussinov Algorithmus

Recursion:
for $n = 2$ **to** L **do** // longer and longer subsequences — O(L)
  for $j = n$ **to** L **do** — O(L)
    $i \leftarrow j - n + 1$

$$\gamma(i,j) \leftarrow \max \begin{cases} \gamma(i+1,j), \\ \gamma(i,j-1), \\ \gamma(i+1,j-1) + \delta(i,j), \\ \max_{i<k<j} \left(\gamma(i,k) + \gamma(k+1,j)\right). \end{cases}$$

— O(L)

Wir haben 3 verschachtelte Schleifen, wobei jede davon O(L) Mal ausgeführt wird.
Daraus ergibt sich, dass die Gesamtzahl der ausgeführten Operationen des Algorithmus $O(L^3)$ beträgt

Lernziele

- Einführung von RNA-Molekülen
- Konzept der RNA-Sekundärstruktur
- Lernen wie eine RNA-Sekundärstruktur grafisch darzustellen

- RNA-Strukturvorhersage
  - **Nussinov algorithmus**
    - Die Motivation
    - Wie dynamische Programmierung in diesem Fall angewendet wird
    - Praktisches Beispiel

Referenzen zu diesem Vortrag

- R. Dubin, S.Eddy, A.Krogh und G. Mitchinson, Biological sequence Analysis, Cambridge, 1998
- Sean R. Eddy: How do RNA folding algorithms work? Nature Biotechnology, Vol 22, Num 11, pages 1457-1458, 2004
- Rune Lyngso, Lecture Notes on RNA Secondary Structure Prediction, 2010