RNA Faltung

Annalisa Marsico

OWL RNA Bionformatics group

Max Planck Institute for Molecular Genetics

Frei Universität Berlin

01/12/2014

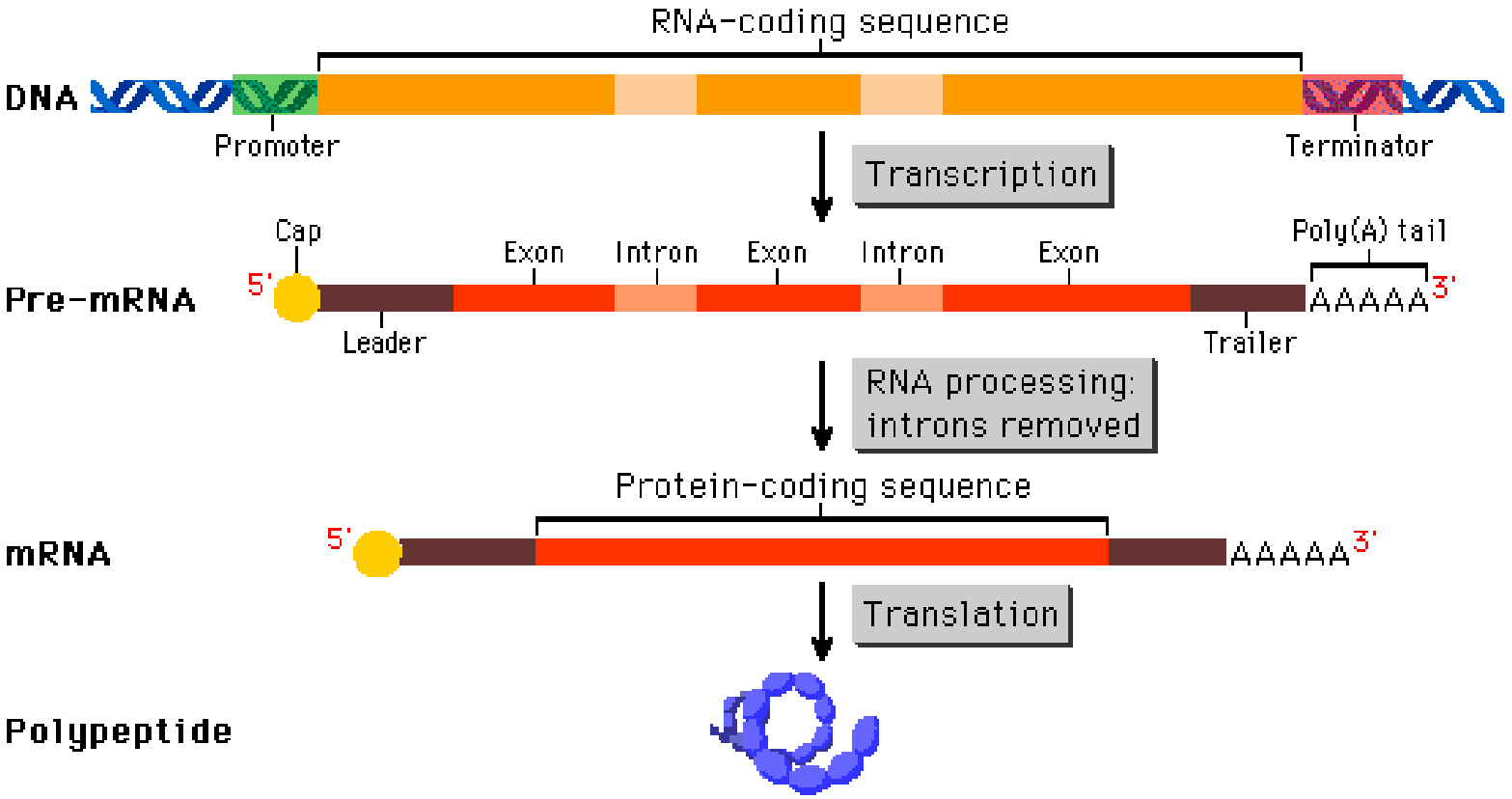
RNA Moleküle

RNA, DNA und Proteine sind die Grundmoleküle des Lebens

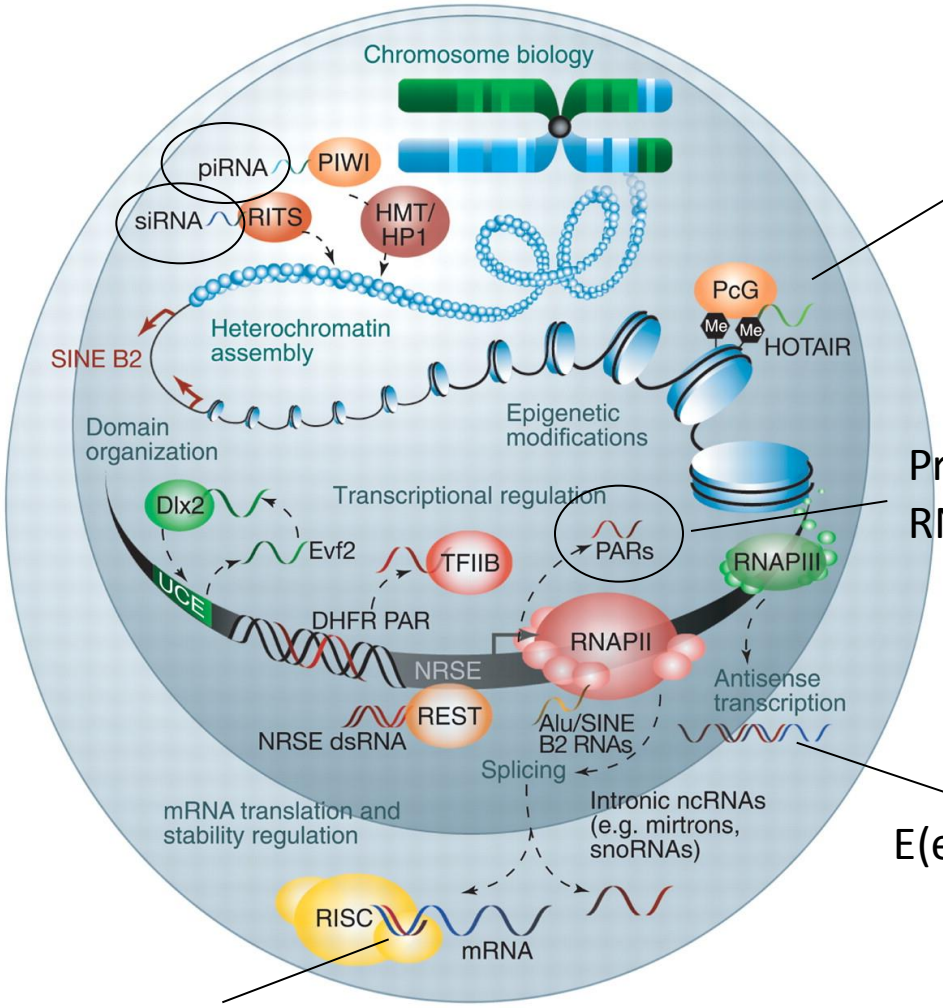
- DNA wird benutzt um die genetische Information zu speichern und replizieren
- Proteine sind die Bausteine der Zelle
- RNAs sind die Zwischenstufen zwischen DNA und Proteinen -> Transkription

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



lincRNAs

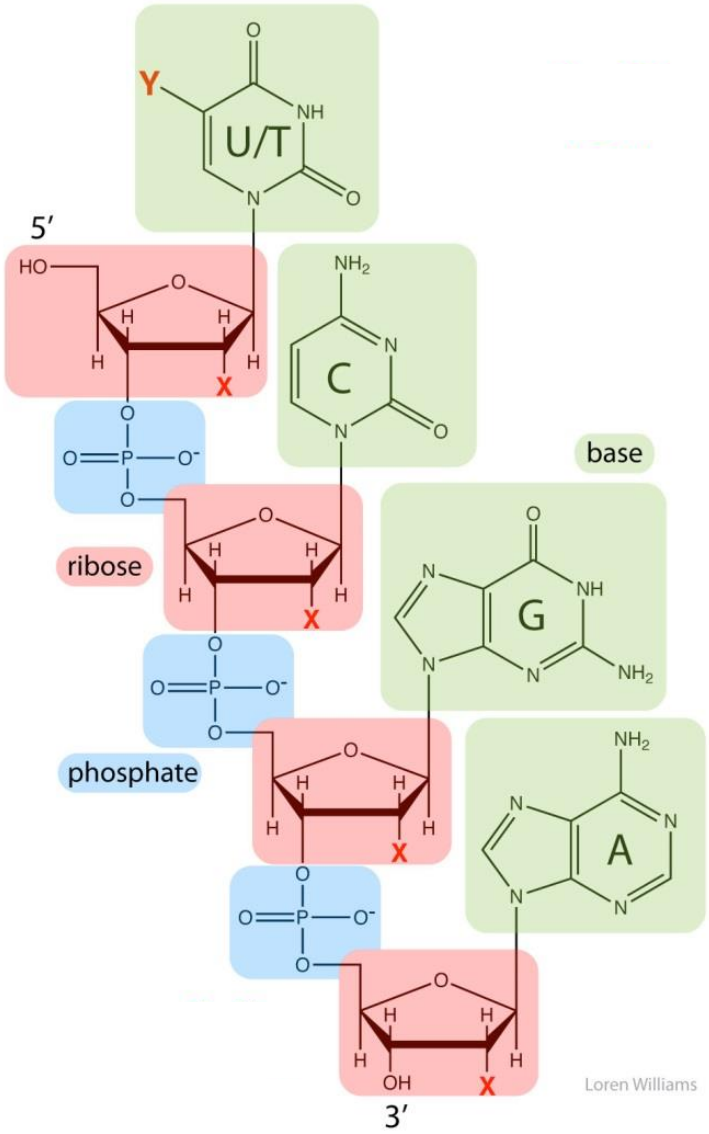
Promoter-associated RNAs

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

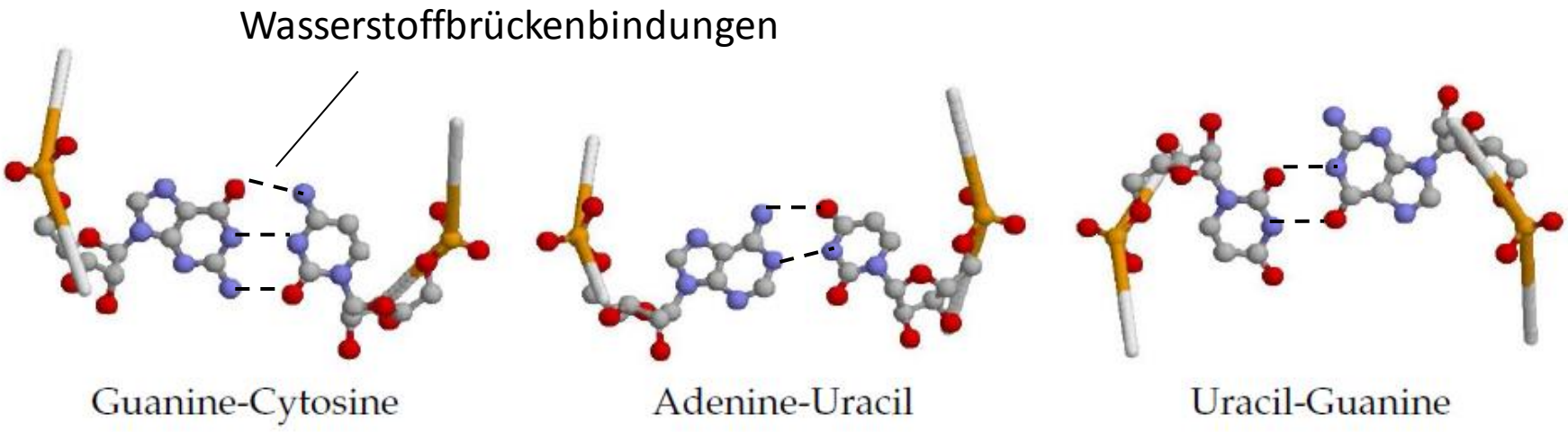
C -> cytosine

G -> guanine

U(T) -> uracil

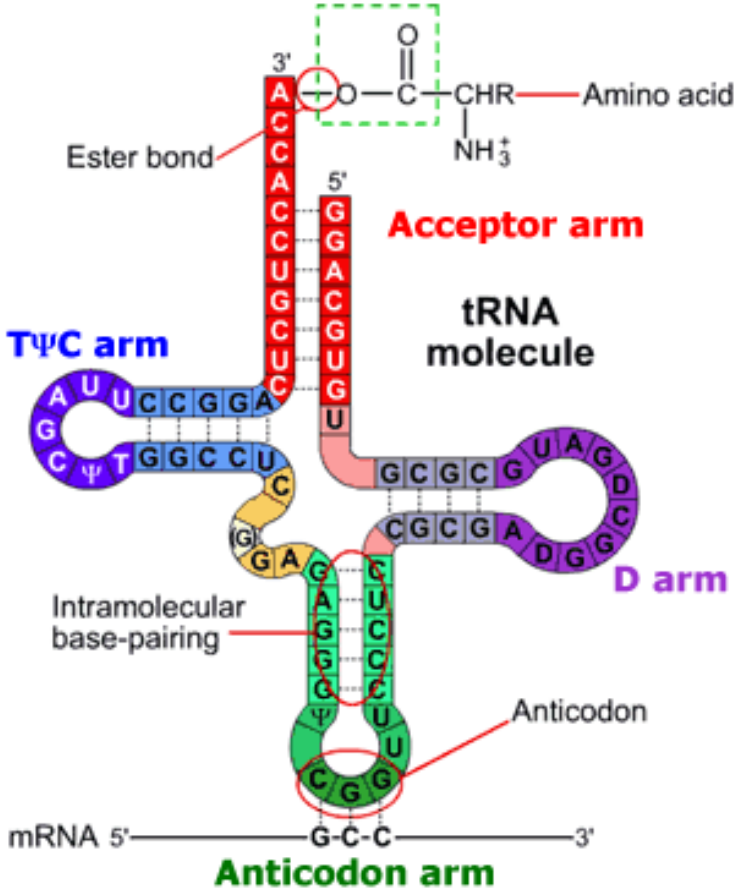
RNA-Sekundärstruktur

Im Gegensatz zu DNA, ist RNA einsträngig. 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

RNA-Sequenzen und Strukturen

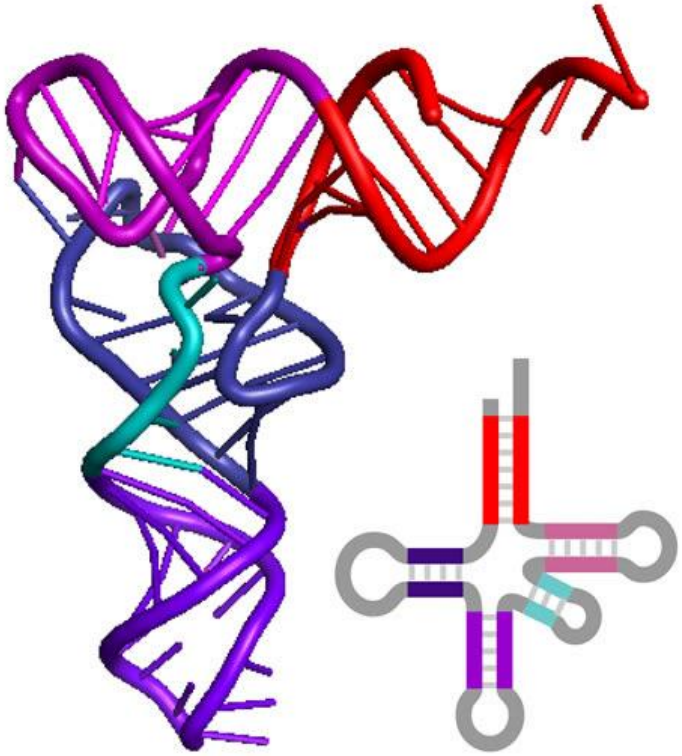
- Die RNA Sequenz faltet sich selbst zurück aufgrund der Komplementarität der Basen.

- 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)

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



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 $x_i \in \{A, C, G, U\}$ für alle *i*

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

1. *j-i* < 3, d.h. die Basenpaaren dürfen nicht zu nahe beieinander liegen, und

2. $\{i, j\} \cap \{i', j'\} = 0$, d.h. die Basenpaare dürfen sich nicht überschneiden.

Formale Definition der RNA-Struktur

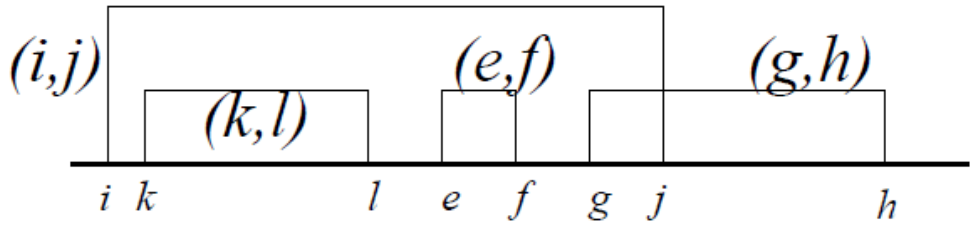
Verschachtelte Strukturen

Definition Eine Sekundärstruktur wird als verschachtelt bezeichnet, wenn:

1. $i < j < i' < j'$, d.h. (i,j) geht (i', j') voran, oder

2. $i < i' < j' < j$, d.h. (i,j) schliesst (i', j') ein

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

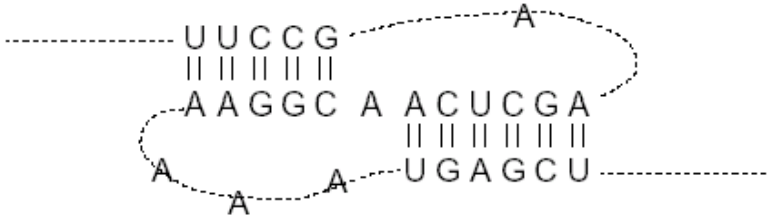


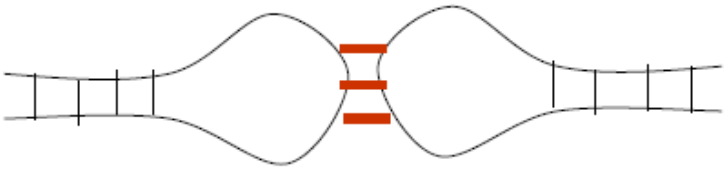
Welche Interaktionen sind hier nicht verschachtelt?

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:

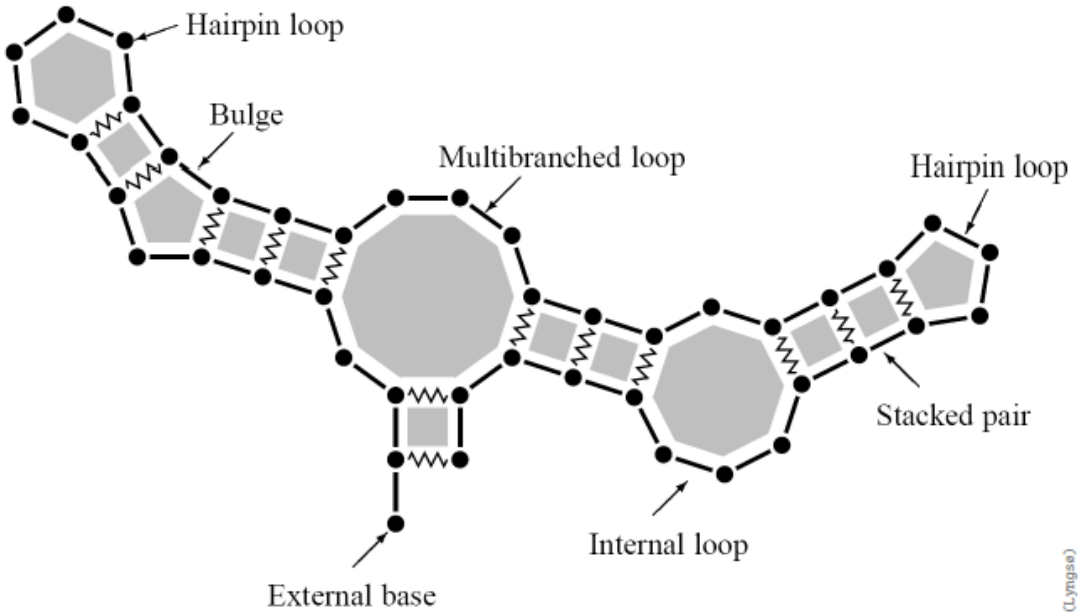
Pseudoknoten





Kissing hairpins

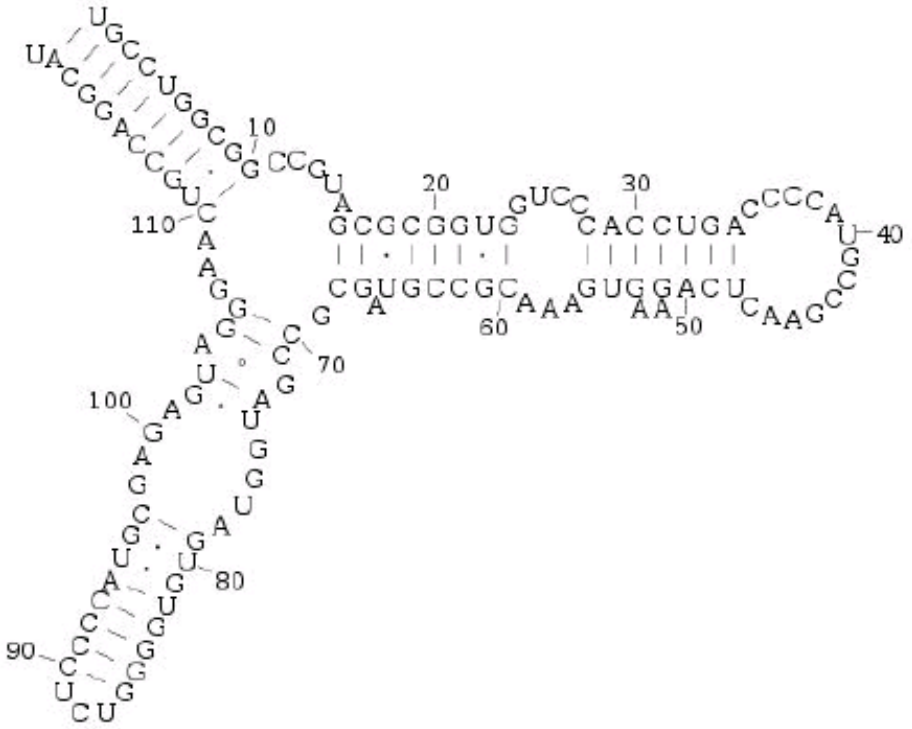
RNA-Sekundärstrukturelemente



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

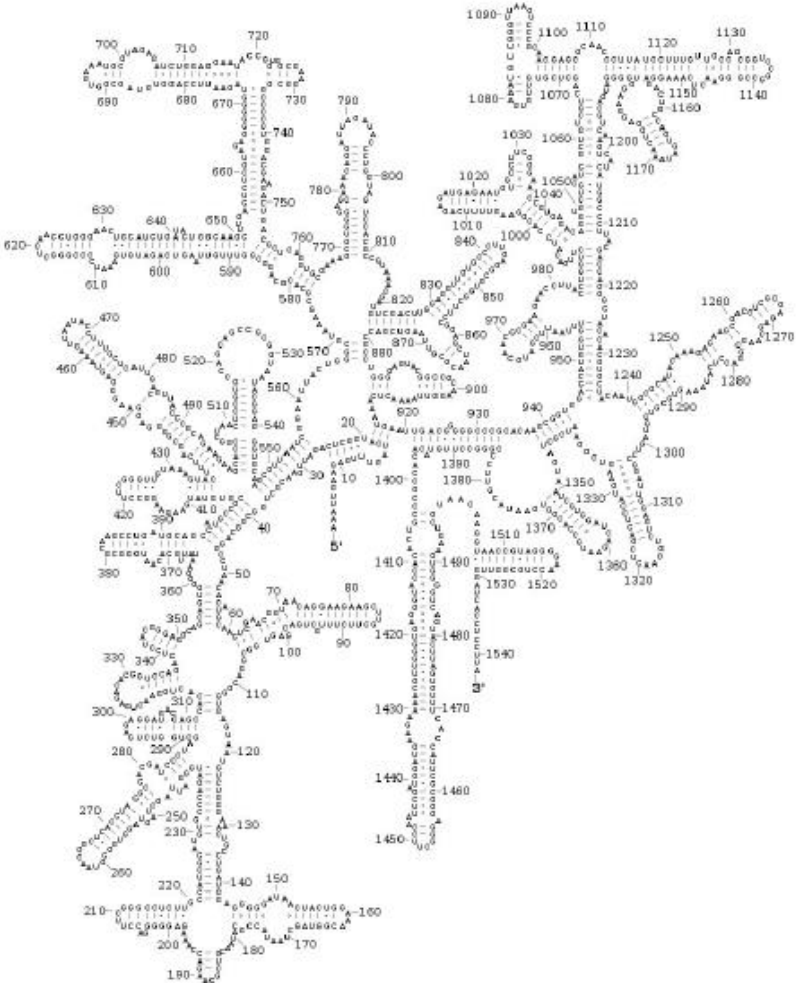
Typen von Einzel- und Doppelstrang-Regionen
Dies nennt man: Basenpaar **Graphdarstellung**

RNA-Struktur Beispiel 1: 5s rRNA



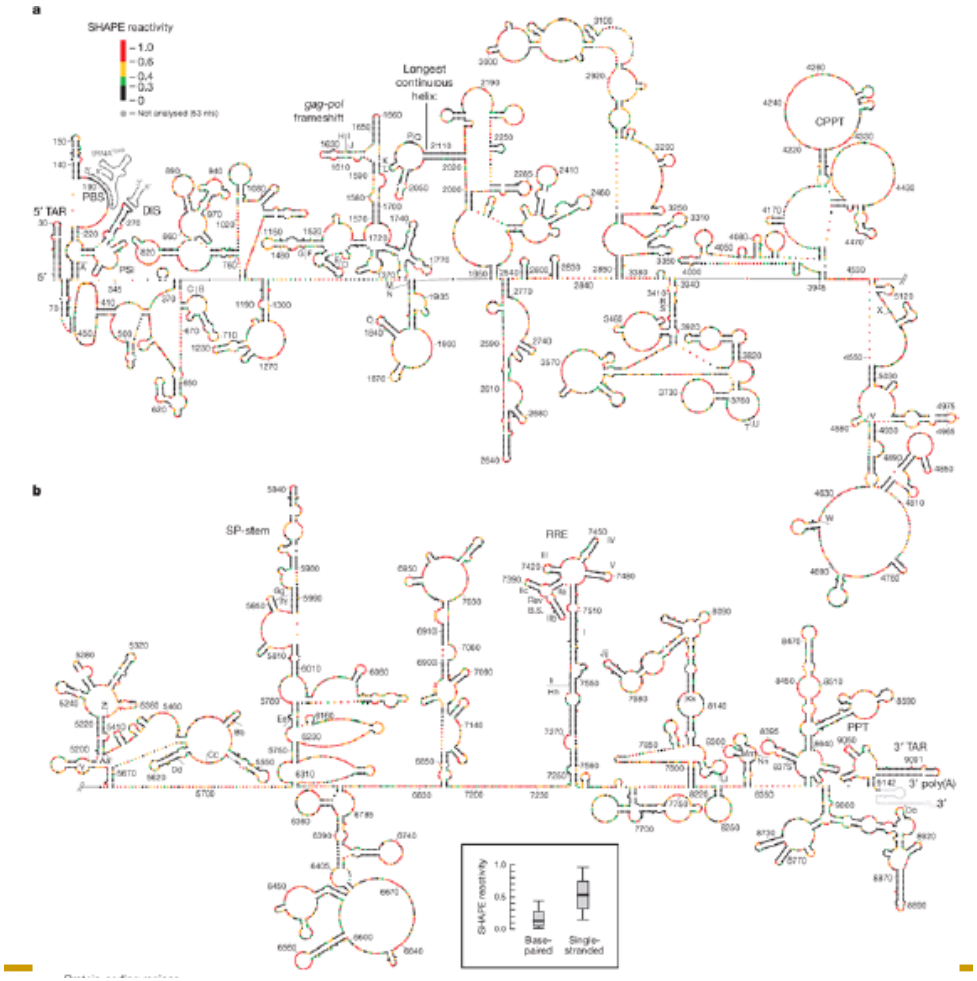
E. coli 5S
120 bases

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



1542 bases

RNA-Struktur Beispiel 1 : HIV

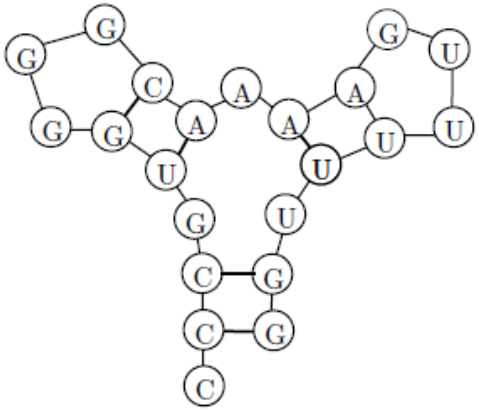


9173 basis

Watts et al., Nature 2013

Darstellungstypen einer RNA-Sekundärstruktur

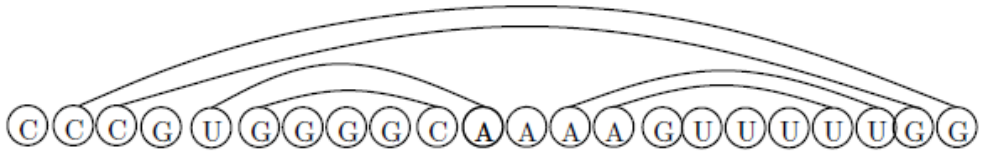
Basenpaar-Graph



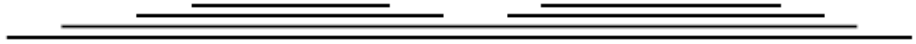
Klammern

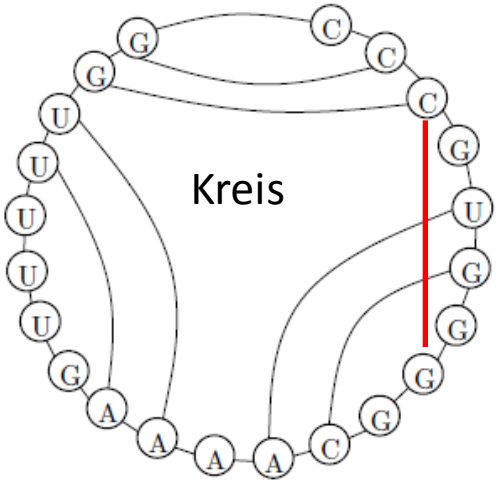
((((***))*((***))*))

Linear

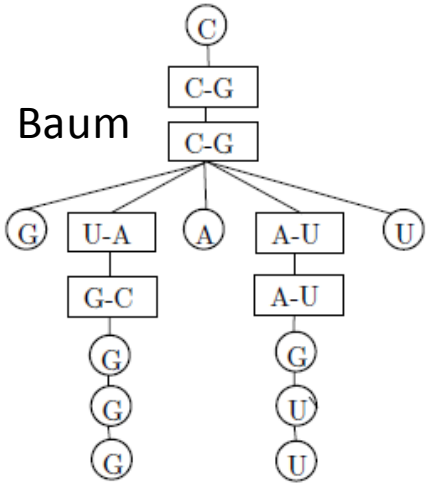


Berg





Baum



Was ist das?

Vorhersage der RNA-Sekundärstruktur

Ansatz:
Finde eine Konfiguration, die die Anzahl von Basenpaaren maximiert
Für eine Sequenz der Länge N, 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 1978 publiziert.

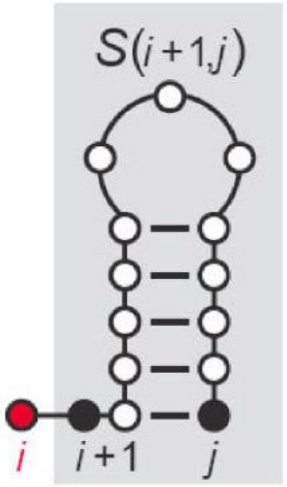
Der Algorithmus ist rekursiv. Es berechnet die beste Struktur für kleinere Teilfolgen größere Sequenzen. Von der kleinsten bis zur vollständigen Sequenz.

Nussinov Faltungalgorithmus -I (1978)

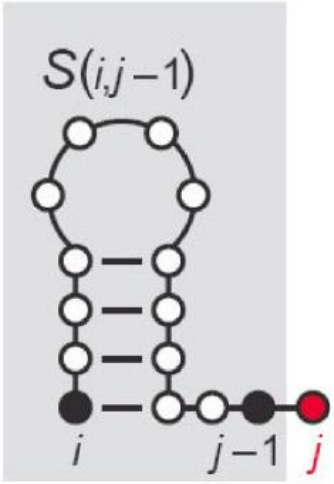
Die Struktur kann rekursiv gefaltet werden

x1…….xL ist eine Sequenz von L zu faltenden Nukletiden. Berechne die maximale Anzahl von gebildeten Basenpaaren der Teilfolge x[i:j]; gegeben das wir bereits für alle kurzen Sequenzen x[m:n] i<m<l<j berechnet haben. Die Struktur auf x[i:j] kann auf vier Arten berechnet werden:

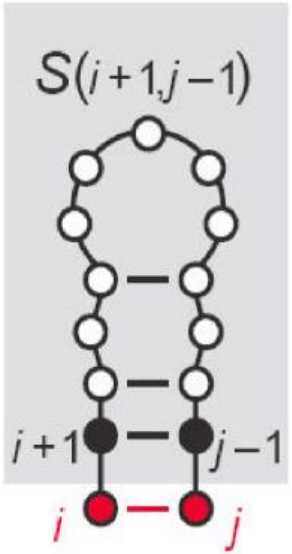
1) Füge eine ungepaarte Base der besten Struktur für Subsequenz [i+1,j] hinzu



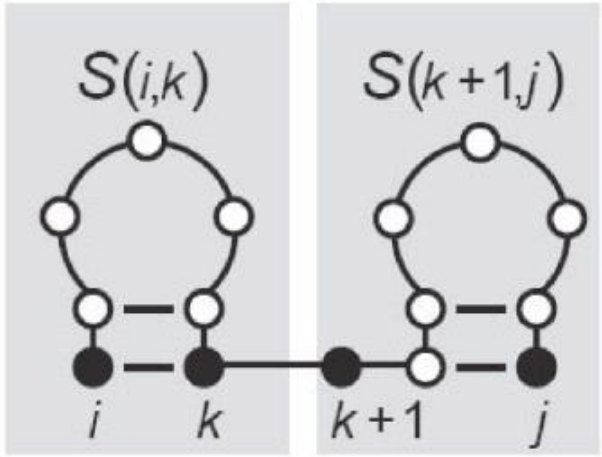
2) Füge eine ungepaarte Base der besten Struktur für Subsequenz [i,j-1] hinzu



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



4) **Bifurkation**: kombiniere zwei optimale Strukturen [i,k], [k+1,j]



Nussinov Faltungalgorithmus -I (1978)

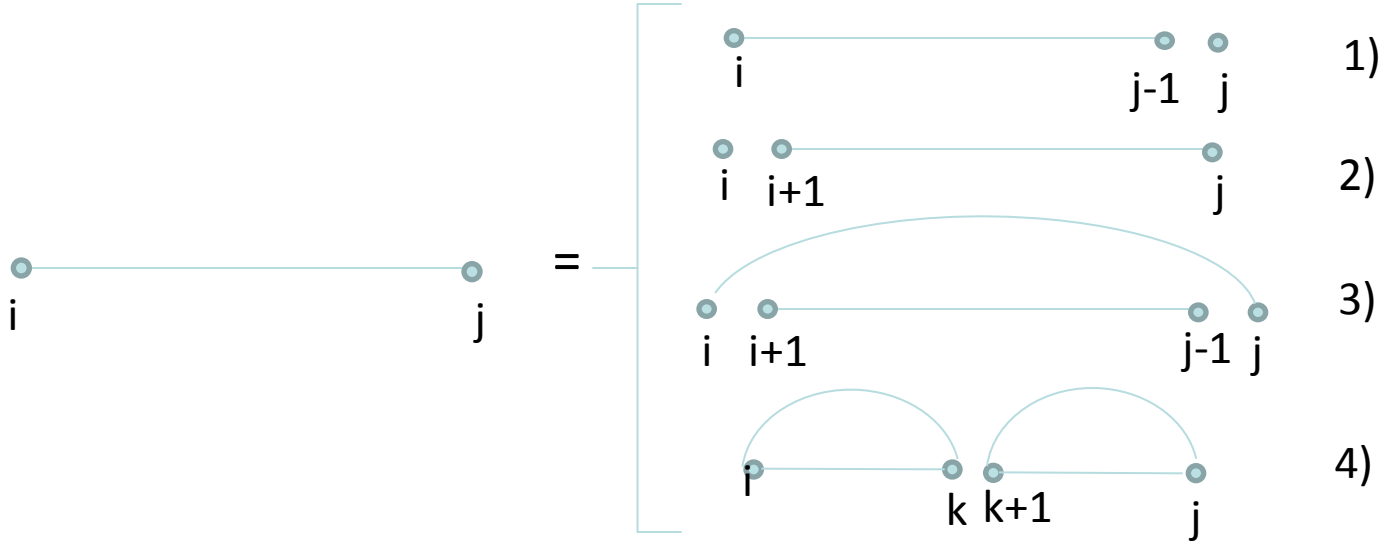
Sei x = (x_1....x_N) eine Sequenz der Länge N. Sei γ(i,j)=1, falls x_i-x_j ein kanonisches Basenpaar ist, sonst 0.

Der Algorithmus der dynamischen Programmierung hat zwei Phasen:

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

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

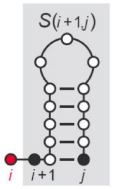
Nussinov Algorithmus – Fill Phase



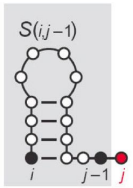
Graphische Darstellung des Faltungsalgorithmus

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

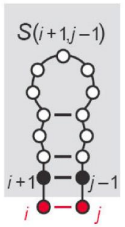
1) i ist ungepaart



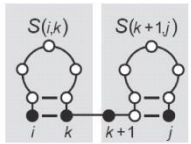
2) j ist ungepaart



3) i,j sind gepaart



4) Bifurkation



Nussinov Algorithmus: Fill Phase

Algorithm (Nussinov RNA folding, fill stage)
Input: Sequence $x = (x_1, x_2, \ldots, x_L)$
Output: Maximal number $\gamma(i, j)$ of base pairs for $(x_i, \ldots, x_j)$.

Initialization:

$$\gamma(i, i-1) = 0 \qquad \text{for } i = 2 \text{ to } L,$$
$$\gamma(i, i) = 0 \qquad \text{for } i = 1 \text{ to } L;$$

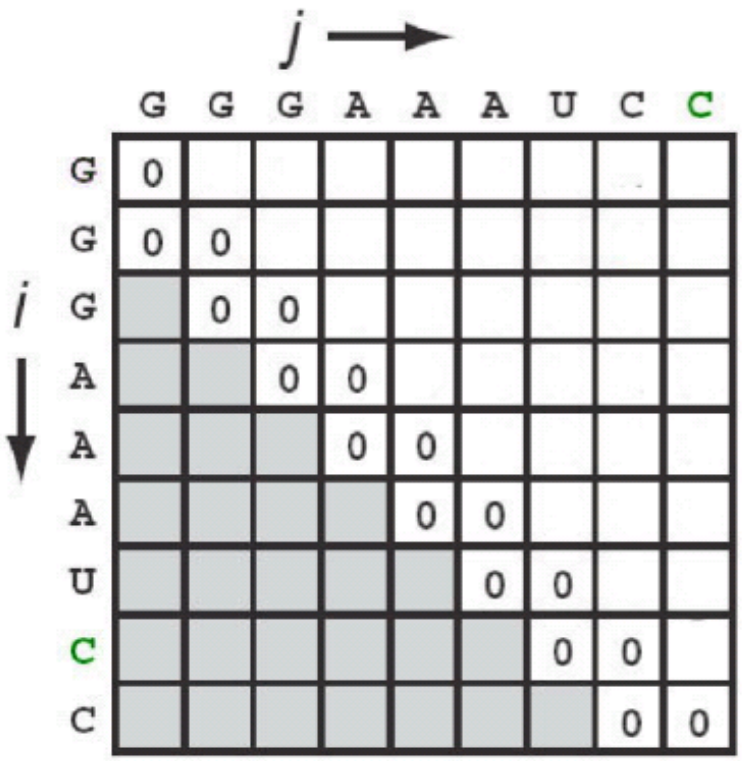
Recursion:

for $n = 2$ **to** L **do** // longer and longer subsequences
  for $j = n$ **to** L **do**
    $i \leftarrow j - n + 1$

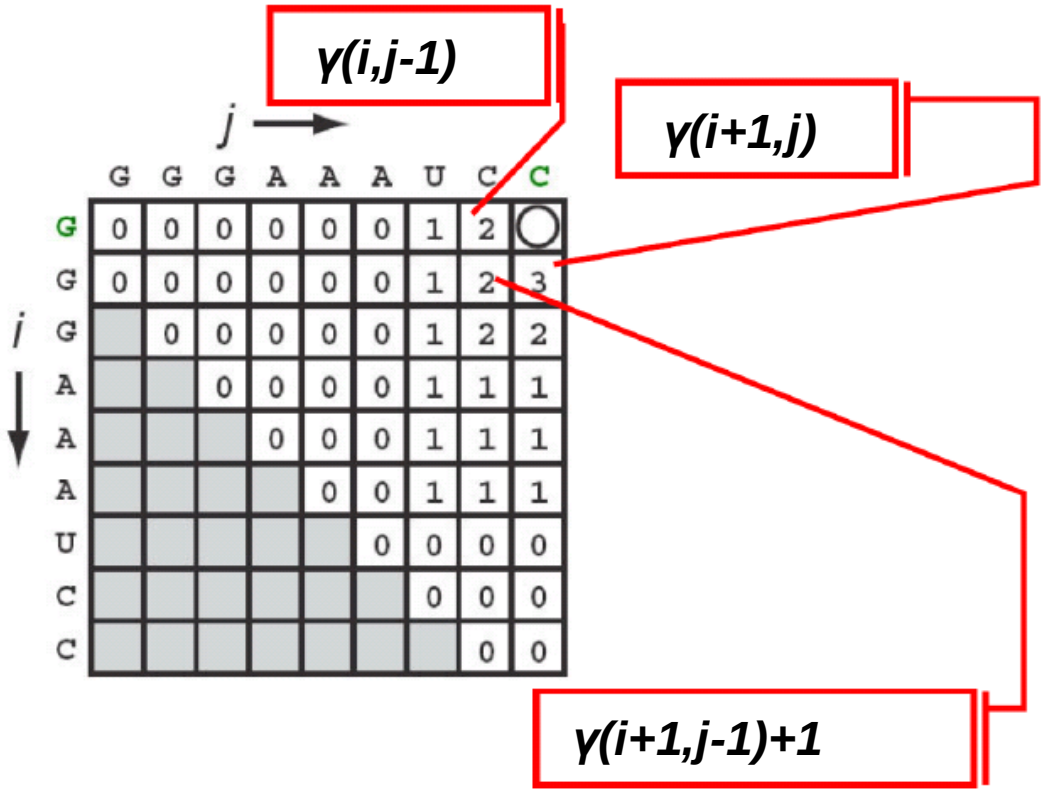
$$\gamma(i, j) \leftarrow \max \begin{cases} \gamma(i+1, j), \\ \gamma(i, j-1), \\ \gamma(i+1, j-1) + \delta(i, j), \\ \max_{i<k<j} \big(\gamma(i, k) + \gamma(k+1, j)\big). \end{cases}$$

Füll die Nussinov Matrix - Beispiel

Sequenz x = GGGAAAUCC. Die Nussinov Matrix γ nach der Initialisierung (i: →, j: ↓)



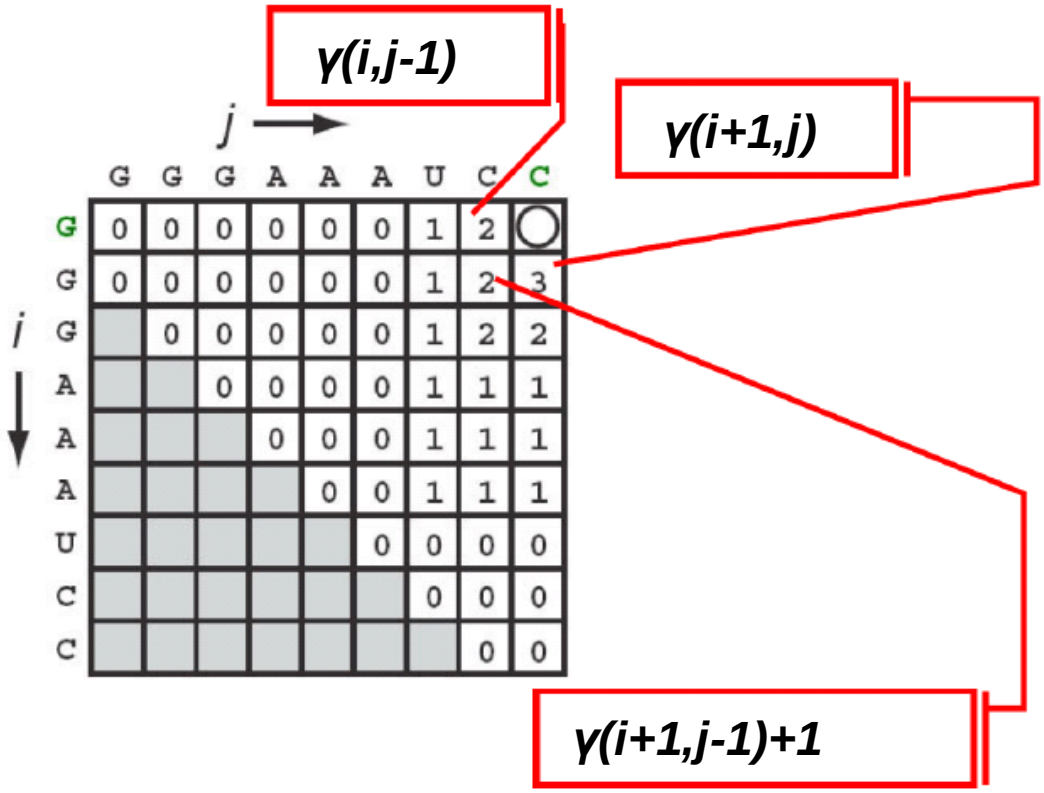
Füll die Nussinov Matrix - Beispiel



Werte werden durch Verwendung des folgenden Bewertungsschemas erhalten

$$\delta(a,b) = \begin{cases} 1 if \ \{a,b\} = \{A,U\} or \ \{C,G\} \\ 0 \ else \end{cases}$$

Füll die Nussinov Matrix - Beispiel



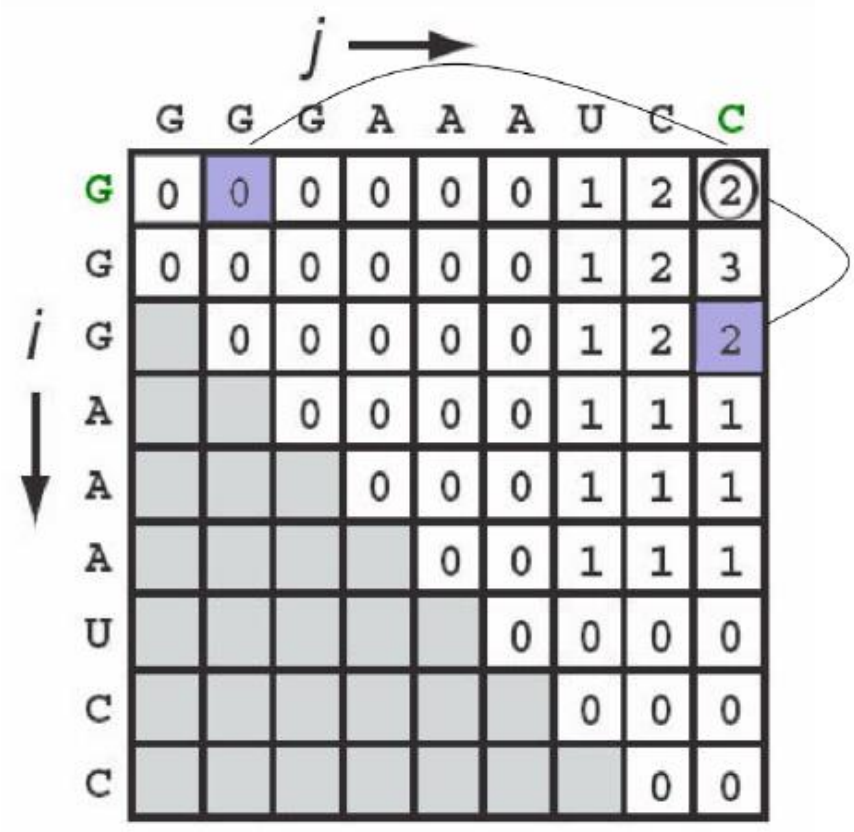
Wir müssen immer noch den Score für Bifurkationen berechnen:
k=2,3,4,5,6,7,8

Füll die Nussinov Matrix - Beispiel

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

Es geht weiter für k=3,4,5,6,7,8

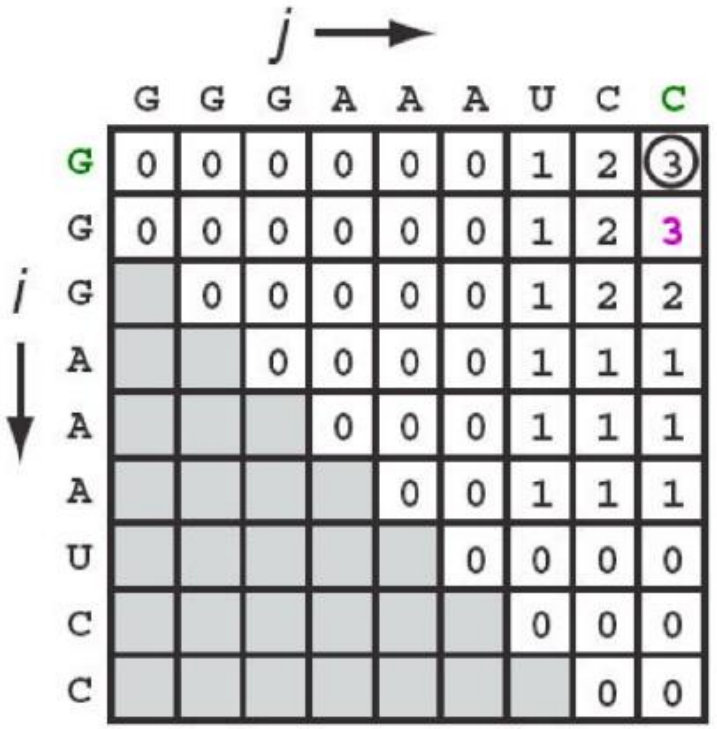
Maximalwert ist 2
erreicht wenn k=2
0+2=2



Füll die Nussinov Matrix - Beispiel

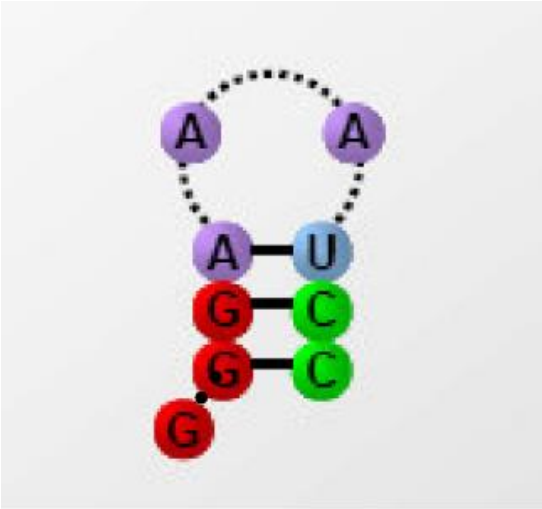
$\gamma(1,9) = \max\{2,3,2,2\}=3$

Traceback, um die Struktur selbst zu finden



Füll die Nussinov Matrix - Beispiel

Bauen die RNA Struktur



		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
i	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

j →

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

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

Nussinov Nachteilen

Die Maximierung der Anzahl an Bp führt unter Umständen nicht zu biologisch relevante Strukturen

- Es gibt verschiedene Moeglichkeiten um Basenpaare zu bilden, Nussinov findet nur A-U und G-C
- Stapeln von Basenpaaren sind nicht berücksichtigt. Dies beeinflusst die Struktur und Stabilität der Helices

G—C G—C
C—G G—C

- Größen von internen Schlaufen sind nicht berücksichtigt

instabil

stabil

instabil

RNA Sekundärstrukturvorhersage: MFE Faltung

- realistischer: Basierent auf Thermodynamik und Statistischer Mechanik
- Die Stabilität einer RNA-Struktur stimmt mit der thermodynamischen Stabilität überein.
- Die MFE ist quantifiziert als die Menge der freien Energie welche durch die Bildung von Basenpaaren freigesetzt wurde

Minimierung der Energie

Freie Energie des Bp(xi,xj) fuhrt zu besseren Ergebnissen

$$E(x, P) = \sum_{(i,j) \in P} e(x_i, x_j)$$

$e(x_i,x_j)$ ist die Menge der freien Energie für „ die Basenpaar (x_i,x_j) “

Die Energiewerte für C-G, A-U and G-U Basenpaare wurden experimentell bei 37 Grad gemessen. Die Werte sind -3, -2, -1 kcal/mol

Nur wenige Änderungen an dem Nussinov Algorithmus würden ihn in ein Minimierungs Problem konvertieren
-> aber die kumulative Wirkung von Stapeln und Schleifen würde dabei nicht beachtet
-> wir brauchen einen besseren Ansatz

Zuker algorithmus: Vorhersage der Sekundärstruktur mit der Thermodynamik

Zuker Algorithmus: die Idee

- Energie Minimierung: Die richtige Struktur ist die eigene mit dem niedrigsten Equilibrium freier Energie
- Man kann die Beiträge für die freie Energie von einzelnen Schleifen messen
- Freie Energien sind additiv. Die Energie einer RNA-Struktur ist die Summe aus den einzelnen Beiträgen.

- Zersetzen die Struktur in Schleifen
  Die Gesamtenergie der Struktur P ist die Summe der einzelnen Beiträge

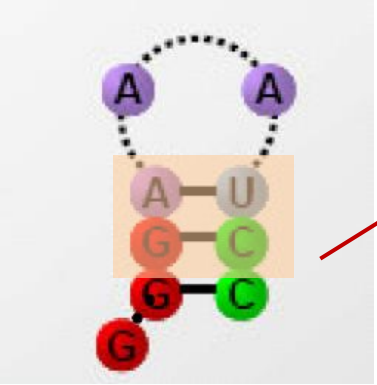
$$E(P) = \sum_{l \in P} E(l)$$

  - E(l) ist der Beitrag einer einzelnen Schleife (Stamm, etc.)

  - **Wichtiger Unterschied zu Nussinov**: Es werden Energien aus Schleifen und nicht von einzelnen Basenpaare gerechnet.

Sequenzabhängige freie Energie

Nächster Nachbar Regel -> Dann müssen wir eine Menge von Regeln definieren, die die Sequenzabhängigkeit beachten



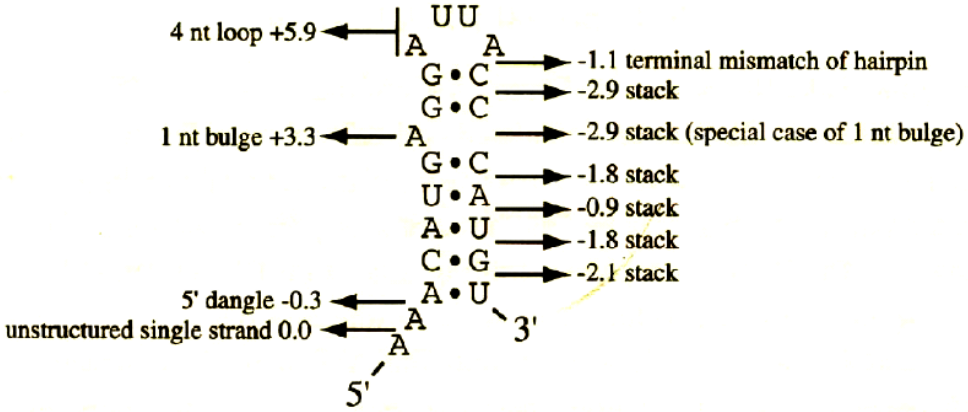
Was ist die freie Energie des GC Basenpaars, wenn AU das Vorherige Basenpaar ist?

Energie wird nur von der vorherigen Basenpaar beinflusst (nicht von Basenpaaren weiter unten).
Der Gesamtenergie = Summe über Stabilität verschiedener Motive / Schleifen
Die Energie wurde experimentell geschätzt aus kleinen synthetischen RNAs

Beispiel Werte: GC GC GC GC
AU GC CG UA
-2.3 -2.9 -3.4 -2.1

Nächster Nachbar Parameter

- Es gibt Schätzungen von ΔG für verschiedene RNA-Strukturmotiven, zB kanonische Paare, Hairpin loops, Beulen, Multi-loops
- Wie sind sie bestimmt?
  - Experimentell: Optical melting Experimente
  - Regeln sind meistens empirisch -> wie sind die in Algorithmen der dynamischen Programmierung implementiert?

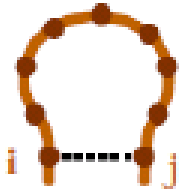


overall ΔG = -4.6 kcal/mol

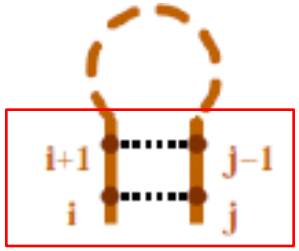
Strukturelemente (formale Definition) (1)

Sei N eine Sequenz und P eine Struktur fur N

- Basenpaar (i,j) in P ist ein **hairpin loop** falls
  i < i‘ < j‘< j fur jeden i,j : (i‘,j‘) ist nicht gepaart (**1-loop**)



- Basenpaar (i,j) in P ist ein **stacking** falls (i+1,j+1) in P ist (**2-loop**)

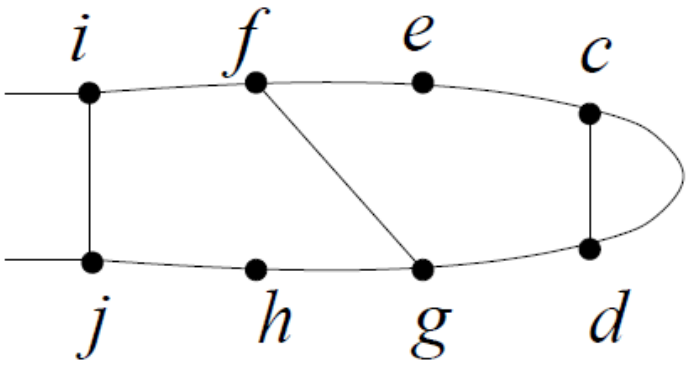


Zuker: k-loop Zerlegung

Wenn (i,j) ein Basenpaar in P ist, dann sagen wir dass

-> **h von (i,j) erreichbar ist**, falls es kein Bp (i‘,j‘) gibt, dass i < i‘ < h < j‘ < j,
sondern gilt i < h <j .

-> Ebenso sagen wir, **ist (f,g) erreichbar von (i,j)** falls f und g es sind.

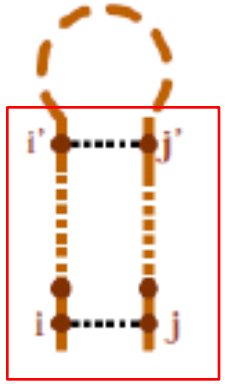


Es gibt k-1 Bp und k‘ ungepaarten Basen, die von (i,j) erreichbar sind.

Die Menge aller möglichen k-1-Basenpaare, die von (i, j) erreichbar sind, wird **k-Loop von (i,j) geschlossen** genannt.

Strukturelemente (formal Definition) (2)

- Basenpaaren (i,j) und (i‘,j‘) in P formen ein **internal loop** falls
  - i < i‘ < j‘ < j
  - (i-i‘)+(j-j‘) > 2 (no stack)
  - es gibt keine Basenpaar (k, l) zwischen (i,j) und (i‘,j‘)

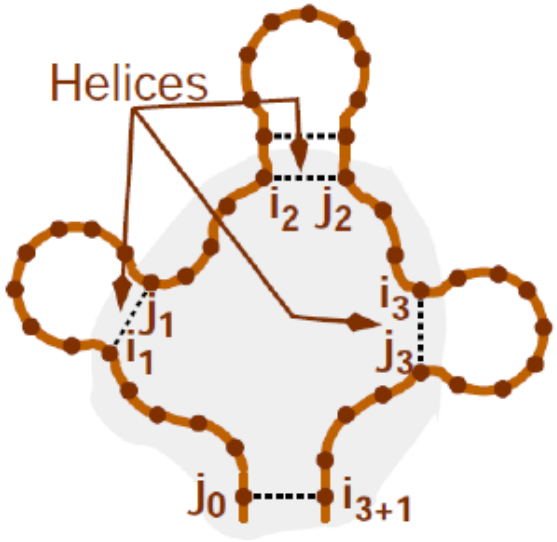


Ein internal loop ist **left (right) bulge genannt**, falls j-j‘>1 oder i‘-i>1 (aber nicht beide)

Strukturelemente (formal Definition) (3)

Ein k-multiloop besteht aus mehreren Basenpaaren $(i_1,j_1)...(i_k,j_k)$ mit einem Schlussbasenpaar (j_0, i_{k+1}) mit der Eigenschaft, dass:

- Fur jeden 0 <= l, l‘ <= k gilt es dass es gibt kein Basenpaar (i‘,j‘) in P mit i‘ in $[j_l,.....i_{l+1}]$ and j‘ in $[j_{l'},.....j_{l'+1}]$
- $(i_1,j_1)...(i_k,j_k)$ werden helices von den multiloop genannt



Ein k-loop mit k >= 3 ist ein **Multiloop oder Multi-branched Loop**

Bemerkung (1)

- Normalerweise müssen hairpin loops mindestens 3 Nukleotiden beinhalten -> für jede hairpin loop (i,j) von P gilt die Bedingung: $i < j-3$

- Jede Sekundärstrukturelement ist durch ihre Abschlussbasenpaar definiert.

Bemerkung (2)

Bei einer Sequenz x = (x_1, x_2, ..., x_N), jedes Sekundärstruktur P auf x teilt die Menge {1.....N} in k-loops S_0,S_1, ...S_m auf.
Jeder k-loop hat hat eine Energie e(s_i) Und die Energie aus einer Struktur P ist

$$E(P) = \sum_{i=0}^{m} e(s_i)$$

Energie ist eine Funktion der k-loops statt eine Funktion der Basenpaare.
Jede e(s_i) wird aus den Nachbarn Basen (nach den Nächster Nachbar Regeln) und den Energiewerten der Schleifen unterschiedlicher Länge berechnet werden

Energiewerte für Stapel und Schleifen

Freie Energien für Basenpaar Stapel

	A/U	C/G	G/C	U/A	G/U	U/G
A/U	-0.9	-1.8	-2.3	-1.1	-1.1	-0.8
C/G	-1.7	-2.9	-3.4	-2.3	-2.1	-1.4
G/C	-2.1	-2.0	-2.9	-1.8	-1.9	-1.2
U/A	-0.9	-1.7	-2.1	-0.9	-1.0	-0.5
G/U	-0.5	-1.2	-1.4	-0.8	-0.4	-0.2
U/G	-1.0	-1.9	-2.1	-1.1	-1.5	-0.4

Freie Energie für Schleifen

size	internal loop	bulge	hairpin
1	.	3.9	.
2	4.1	3.1	.
3	5.1	3.5	4.1
4	4.9	4.2	4.9
5	5.3	4.8	4.4
10	6.3	5.5	5.3
15	6.7	6.0	5.8
20	7.0	6.3	6.1
25	7.2	6.5	6.3
30	7.4	6.7	6.5

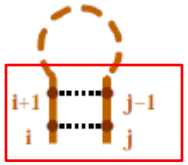
Schleifen-abhängige Energien

- **Hairpin loop** (i,j)

  eH(i,j)

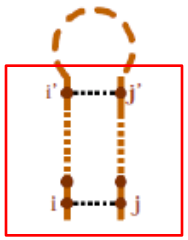
- **Stacking** (i,j)

  eS(i,j,i+1,j+1)



- **Internal loop** (i,j,i‘,j‘)

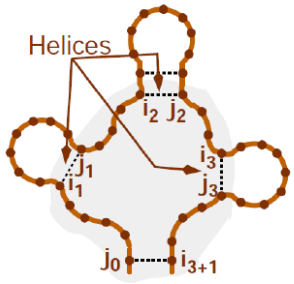
  eL(i,j,i‘,j‘)



- **K-multiloop**

  eM(j_0,i_1,j_1, ….i_k,j_k,i_{k+1})

  Berechnung für alle möglichen k-loops



Zuker Algorithmus - Die Matrizen

- Sei S ein RNA Molekül der Länge N; sei i und j zwei Nukleotiden von S mit $1 \leq i \leq j \leq N$
- Für alle Basenpaaren i,j $1 \leq i \leq j \leq N$ sei **W(i,j)** die minimale freie Energie aller Strukturen, die durch die Subsequenz $S_{i,j}$ geformt werden
- **$V_{(i,j)}$** minimale freie Energie aller Strukturen, die durch die Subsequenz $S_{i,j}$ geformt werden, in denen i und j zueinander gepaart sind
- **$WM_{(i,j)}$** minimale freie Energie aller Strukturen, die durch die Subsequenz $S_{i,j}$ geformt werden, welche Teil einer Multischleife sind

Zucker algorithmus – Rekursionschritt

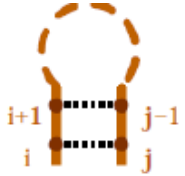
$$W_{i,j} = \min \begin{cases} W(i+1, j) \\ W(i, j-1) \\ V(i, j) \\ \min\limits_{i<k<j} \{W(i,k) + W(k+1, j)\} \end{cases}$$

$$V(i, j) = \min \{E_1, E_2, E_3, E_4\}$$

Die minimale Faltungsenergie E_{min} ist gegeben durch W(1,N)

Zuker: loop Zerlegung, rekursive Berechnung von V(i,j)

⇒ Hairpin loop $V(i,j) = eH(i,j)$ **Fall 1**

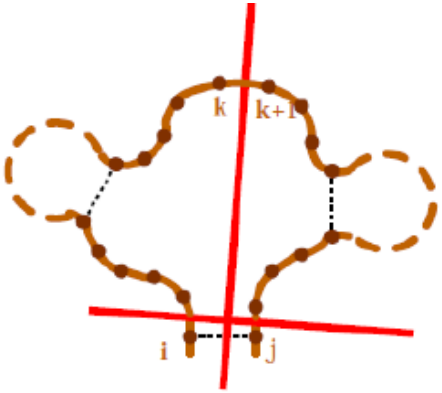


⇒ Stapeln $V(i,j) = eS(i,j,i+1,j-1) +$

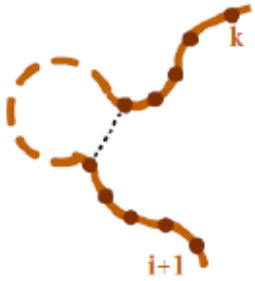
Fall 2

⇒ Innere Schleife $V(i,j) = eL(i,j,i',j') +$

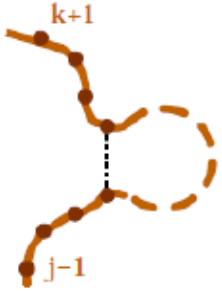
Fall 3



⇒



+



+ a

Strafe für das Öffnen eines multiloop

Fall 4

$VM_{i+1,k}$ + $VM_{k+1,j-1}$ + a

Zuker: loop Zerlegung, rekursive Berechnung von V(i,j)

- Zusammenfassend, V(i,j) ist die minimale freie Energie, die in vier Wegen berechnet werden kann:

$$V(i,j) = \min \{ E_1, E_2, E_3, E_4 \}$$

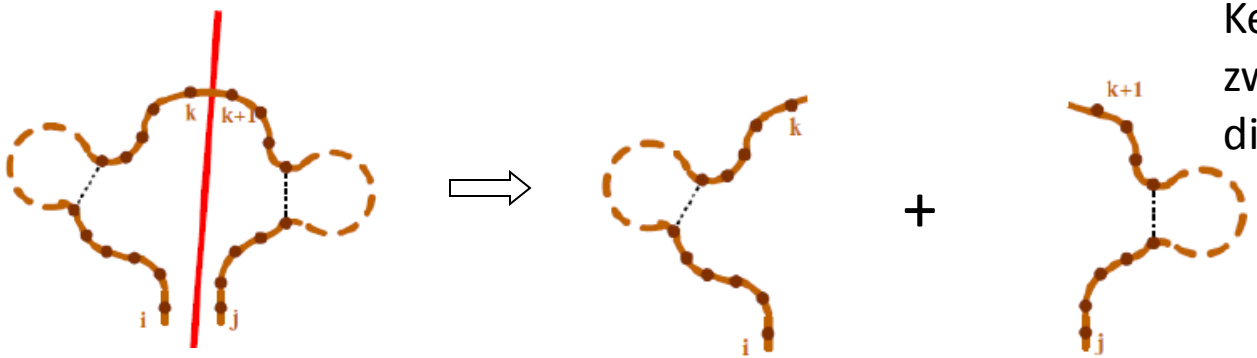
$$E_1 = eH\ (i,\ j)$$

$$E_2 = eS\ (i,\ j) + V\ (i+1,\ j-1)$$

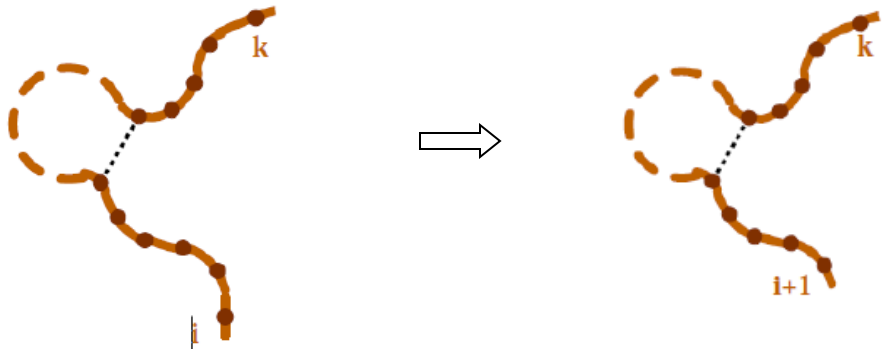
$$E_3 = \min_{i<i'<j'<j} \left[eL\ (i,\ j, i',\ j') + V\ (i',\ j') \right] \equiv VBI\ \ (i,\ j)$$

$$E_4 = WM\ \ (i,\ j)$$

Zuker: multiloop Behandlung, rekursive Berechnung von VM(i,j)



Keine Basenpaare zwischen i und j, setze die Loop Zerlegung fort



Ein baumeldes Ende (D.h. ungepaarte Nukleotiden)

c

Strafe für ungepaarte Nukleotiden

$V(i, j)$ + b

i und j formen ein Basenpaar

Strafe für 'geschlossene' Strukturen im multiloop

Zuker: multiloop Behandlung, rekursive Berechnung von MW(i,j)

- WM(i,j) -> S_i, ...S_j ist Teil eines Multiloop (i,j kein Basenpaar!)
- Der Multiloop muss mindestens einmal geteilt werden, sonst ist es ein einfacher loop
- **Idee: schneide Teile des Multillops bis nur einzelnen hairpin loops übrig sind**

$$VM(i,j) = \min \begin{cases} VM(i+1,j) + c \\ VM(i,j-1) + c \\ \min\limits_{i \le k \le j} \left[VM(i,k) + VM(k+1,j) \right] \\ (V(i,j) + b) \end{cases}$$

Zusammenfassung der Rekursionen des Zuker Algorithmus

$$W_{i,j} = \min \begin{cases} W(i+1, j) \\ W(i, j-1) \\ V(i, j) \\ \min\limits_{i<k<j} \{W(i,k) + W(k+1, j)\} \end{cases}$$

die minimale freie Energie aller Strukturen,
die durch die Subsequenz $S_{i,j}$ geformt werden

i,j gepaart

$$V(i,j) = \begin{cases} \min \begin{cases} eH(i,j) & \text{hairpin} \\ eS(i,j) + V(i+1, j-1) & \text{stack} \\ VBI(i,j) & \text{Internal loop} \\ VM(i,j) & \text{multiloop} \end{cases} \\ \infty \end{cases}$$

Paar (i,j) nicht erlaubt

$$VM(i,j) = \min \begin{cases} VM(i+1, j) + c \\ VM(i, j-1) + c \\ \min [VM(i,k) + VM(k+1, j)] \\ V(i,j) + b \end{cases}$$

Alle Möglichkeiten um
ein Multiloop zu zersetzen

$$VBI(i,j) = \min_{i<i'<j<j'} [eL'(i, j, i', j') + V(i', j')]$$

Alle Möglichkeiten um
ein Internal loop zu formen

Die minimale Faltungsenergie E_{min} ist gegeben durch W(1,N)

Zeitanalyse

Die Berechnung von:

- W dauert $O(L^3)$ Schritte
- V dauert $O(L^2)$ Schritte
- E_3 dauert $O(L^4)$ Schritte
- WM dauert$O(L^3)$ Schritte

Die gesamte Laufzeit (run-time) ist $O(L^4)$

Der praktischste Weg um die Laufzeit auf **$O(L^3)$ zu** reduzieren ist die Grösse der inneren Loops auf einige konstante Werte zu begrenzen.

Zusätzlich, ist die Energie eines Multiloop nich konstant, sondern ändert sich linear:
E(multi_loop)=a+bk‘+ck
k‘ = Anzahl der ungepaarten Basen
k = Anzahl der Helixen

Komparative RNA Analyse

Die Nussinov und Zucker Algorithmen berechnen die Beste RNA Faltung durch die Optimierung einer bestimmten objectiven Funktion.

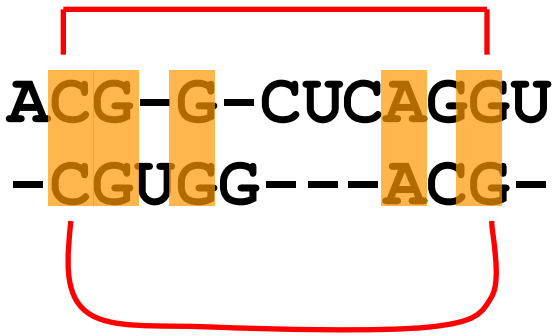
Obwohl Energieminimierungstechniken attraktiv sind, zuzeit funktionieren fast alle die RNA-Sekundärstrukturvorhersage Methoden mit komparativer Analyse. Allerdings, brauche komparative Methoden viele unterschiedliche Sequenzen als Input und hochwertigen multiplen Alignments gut zu funktionieren.

Ein erfolgreicher Ansatz basiert auf den Vergleich der Sequenz-Struktur von RNAs.

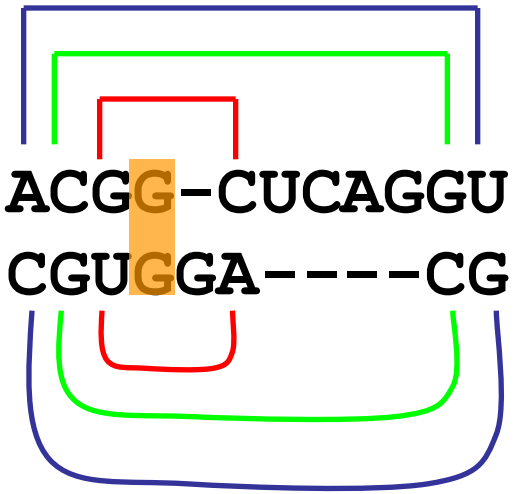
Problem: erfasse die Sekundärstrukturkonservierung und möglichst auch die Sequenzkonservierung

Komparative RNA Analyse

Z.b. Hier zwei RNA Alignments: eines bewahrt nicht die sekundäre Struktur, während das andere diese bewahrt (und damit bewahrt die Interaktionen)



Alignment mit
hohem Sequenz Score
-> Sekundärstruktur
nicht erhalten



Alignment mit
hohem Interaktions Score
-> Sequenz nicht erhalten

RNA Faltung mit komparativer Analyse

Die **Schlüsselidee** besteht darin, die **Interaktionen** zu identifizieren (das ist der Watson-Crick korrelierten Positionen) in einem Multiple Alignment und Sie verwenden um die Sekundärstruktur zu vorhersagen
Mutual information content (gemeinsamer Informationgehalt)

Das ist ein eingegebene MSA. Welche sind die konservierte Interaktionen (die keine Sequenzkonservierung erhalten)?

```
Seq1   GCCUUCGGGC
Seq2   GACUUCGGUC
Seq3   GGCUUCGGCC
Seq4   GACUUUGGUC
```

RNA Faltung mit komparativer Analyse

```
Seq1    GCCUUCGGGC
Seq2    GACUUCGGUC
Seq3    GGCUUCGGCC
Seq4    GACUUUGGUC
```

Die Korrelation zwei Positionen kann als gemeinsamer Informationsgehalt Maß berechnet werden:
„Wenn Ihr mir sagt, die Identität der Position i, wie viel kann ich über die Identität der Position j lernen? “

Mutual Information

- Eine Methode zu lokalizieren kovariante Positionen in einem MSA ist **Mutual Information** von zwei Spalten.
- Formel zur Berechnung des mutual Information Content:

$$MI_{i,j} = \sum_{x,y} f(x_i, y_j) \log_2 \frac{f(x_i, y_j)}{f(x_i) f(y_j)}$$

- $f(x_i)$: Frequenz jedes Base x in {A,C,G,U} in Spalte i des Alignments
- $f(x_i,y_j)$: zusammengeführte Frequenz zweier nukleotide x in Spalte i und y in Spalte j
- Ratio

MI in dynamic programming algorithms

$$W_{i,j} = \min \begin{cases} W(i+1, j) \\ W(i, j-1) \\ V(i, j) - MI(i, j) \\ \min\limits_{i<k<j} \{W(i,k) + W(k+1, j)\} \end{cases}$$

Referenzen zu diesem Vortrag

- R. Dubin, S.Eddy, A.Krogh und G. Mitchinson, Biological sequence Analysis, Cambridge, 1998
- **M. Zuker** and P. Stiegler: Optimal computer folding of large RNA sequences using thermodynamics

  Nature Biotechnology, Vol 22, Num 11, pages 1457-1458, 2004
- Rune Lyngso, Lecture Notes on RNA Secondary Structure Prediction, 2010