

Bioinformatik von RNA- und Proteinstrukturen

Inhaltsverzeichnis

1	Formale Sprachen	1
1.1	formale Grammatik G	1
1.2	Klassifikation von formalen Sprachen	1
1.3	Hidden Markov Model	3
2	Einleitung	5
2.1	RNA	5
2.2	R/DNA-Sekundärstruktur	6
2.3	Strukturabbildungen	7
3	Strukturvorhersage	10
3.1	Nussinov	10
3.2	Turner-Modell (Nearest-Neighbor-Modell)	11
3.3	Zuker-Algorithmus	11
3.3.1	suboptimales Falten	11
3.4	Wuchty-Algorithmus	11
3.4.1	Wuchty-Backtracking	11
3.5	McCaskill	11
3.6	stochastisches Backtracking	11
3.7	Strukturvorhersagen verbessern	11
3.8	Konsensusstrukturvorhersagen	11
3.9	Wie kann RNA evolvieren?	11
3.9.1	Neutrale Netzwerke	11
3.9.2	SHAPE-Abstraktion	11
3.9.3	Energielandschaften	11
3.9.4	Faltungskinetik	11
3.9.5	Barriers Trees	11
3.9.6	Flooding-Algorithmus	11
3.9.7	Co-transcriptional folding	11
4	weitere Bindungsarten, erlaubte Basenpaare	12
5	Proteine	13
6	Sekundärstrukturelemente	14
6.1	Chou-Fasman (Sekundärstrukturvorhersage von Proteinen)	14
7	(Protein-) Strukturvorhersage (3D)	17
7.1	Strukturaufklärung	17
7.2	Qualität der Strukturvorhersage	17
7.3	Problem der Strukturvorhersage (Levinthal-Paradoxon)	17
7.4	Protein-Domains (Domänen)	18
7.5	Zwei Typen von Vorhersagen	19
7.5.1	Ab-initio-Vorhersage	19

7.5.2	Template based methods	22
-------	----------------------------------	----

1 Formale Sprachen

Formale Sprache¹ L über Alphabet Σ

$L \subseteq \Sigma^*$

mit Σ^* = Kleensche Hülle² von Σ

$$\Sigma^* = \bigcup_{n=0}^{\infty} \Sigma^n$$

$\Sigma^0 = \{\varepsilon\}, \Sigma^1 = \Sigma, \Sigma^2 = \Sigma \times \Sigma$

$\varepsilon \rightarrow$ leeres Wort (leere Menge)

Beispiel: $\Sigma = \{a\}, \Sigma^* = \{\varepsilon, a, aa, aaa, \dots\}, L = \{a, aa, aaaa, \dots\}$

1.1 formale Grammatik G

$G = (N, \Sigma, P, S)$ mit

- N = Nichtterminale
- Σ = Alphabet
- P = Produktionsregeln
- S = Startsymbol ($\in N$)

$P \subseteq (N \cup \Sigma)^* / N (N \cup \Sigma)^* \rightarrow (N \cup \Sigma)^*$

Beispiel:

$G = (\{S\}, \{a\}, \{S \rightarrow aaS, S \rightarrow a\}, S)$

führt zu: $S \rightarrow aaS \rightarrow aaa$

1.2 Klassifikation von formalen Sprachen

durch die Comsky-Hierarchie³:

- Typ 0 = rekursiv auszählbar ($\alpha N \beta \rightarrow \gamma$)
- Typ 1 = kontext-sensitiv ($\alpha N \beta \rightarrow \alpha \gamma \beta$)
- Typ 2 = kontext-frei, $N \rightarrow (N \cup \Sigma)^* \rightarrow$ stochistisch kontextfreie Grammatik (SCFG) \rightarrow Dynamics Programming

¹https://de.wikipedia.org/wiki/Formale_Sprache

²https://de.wikipedia.org/wiki/Kleenesche_und_positive_H%C3%BClle

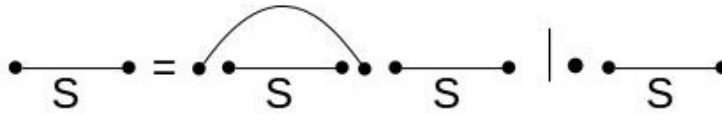
³<https://de.wikipedia.org/wiki/Chomsky-Hierarchie>

- Typ 3 = regular ($N \rightarrow \Sigma | \Sigma N$) \rightarrow dann immer Hidden Markov Model (HMM) modellierbar

bei Alignments: $\boxed{S} \longrightarrow \boxed{S} \begin{smallmatrix} \vdots \\ \vdots \end{smallmatrix} \mid \boxed{S} \text{---} \mid \boxed{S} \text{---} \mid \varepsilon$

Erweiterung mit Wahrscheinlichkeit: $G=(N, \Sigma, P, S, \Omega)$
mit Ω = Wahrscheinlichkeit für Produktionsregeln

jetzt auf RNA-Vorhersagen:

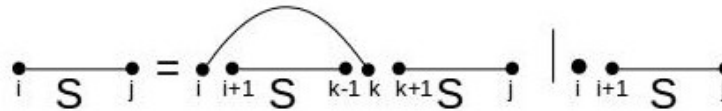


scoring scheme: Bewertung von $\sigma(\curvearrowright) = 1$, $(\sigma(\text{---}))$, $\sigma(\bullet) = 0$
scoring function:

- max Basepairs: + (Summe),
- Anzahl der Strukturen: \cdot (Multiplikation)

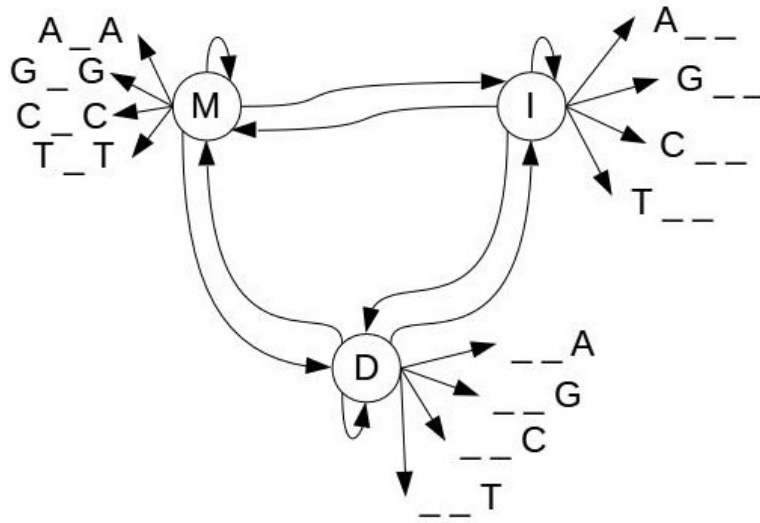
choice function:

- max Basepairs: max,
- Anzahl der Strukturen: + (Summe)



$$S_{ij} = \begin{cases} S_{i+1,j} + \sigma(\bullet) \\ S_{i+1,k-1} + S_{k+1,j} + \sigma(\curvearrowright) \end{cases}$$

1.3 Hidden Markov Model



M: Match, I: Insertion, D: Deletion

Grammatik:

- $M \rightarrow M_{A_A} | \dots | I | D$
- $I \rightarrow I_{A_{--}} | \dots | D | M$
- $D \rightarrow D_{--A} | \dots | M | I$

Beispiel:

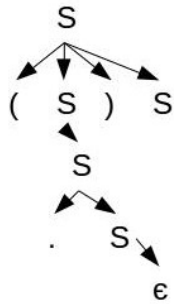


Faltungsgrammatik

$S \rightarrow (S)S | .S | \varepsilon$

Nichtterminale = S, Alphabet = $\{ (,), . \}$

Beispiel in Baumdarstellung:



weiteres Beispiel: Sankoff, Kombination von zwei Grammatiken (Alignment und Faltung)

Alignmentgrammatik

$$S \rightarrow .S|_S|\varepsilon$$

$$G = (N = \{S\}, \Sigma = \{., _\}, P = \{S \rightarrow .S|_S|\varepsilon\}, S)$$

$$\text{Alignment: } G^2 = G \times G = (N \times N, \Sigma \times \Sigma, P^2, (S, S))$$

$$P^2 = P \times P = \begin{pmatrix} S \\ S \end{pmatrix}$$

2 Einleitung

Struktur: Form \rightarrow Funktion

Funktion folgt Form, Form folgt Sequenz

Proteine, RNA, DNA: Sequenzen

4 Strukturlevels:

- primäre Struktur (Sequenz): 1 Dimension
- sekundäre Struktur (grobe Annäherung an Struktur): 2 Dimensionen
- tertiäre Struktur (räumliche Struktur): 3 Dimensionen
- quartäre Struktur (räumliche Anordnung von interagierenden Strukturen): 4 Dimensionen

Behandlung hauptsächlich 2D

2.1 RNA

⁴ Funktion:

- Informationsträger
- Regulator/Katalysator
- Theorieder RNA-World

- Nicht-Messenger-RNA: ncRNA (nc - non-coding)

- Aufbau: Zucker-Phosphat-Rückgrat
- Basen:
 - Purine: Adenin, Guanin
 - Pyrimidine: Cytosin, Uracil
- Paarung: A-U, G-C
- RNA einzelsträngige A-Helix (DNA: doppelsträngige B-Helix)

⁴<https://de.wikipedia.org/wiki/Ribonukleins%C3%A4ure>

2.2 R/DNA-Sekundärstruktur

Definition: Liste von Basenpaaren, sodass gilt (theoretische Regeln):

- erlaubte Basenpaarungen:
 - Watson-Crick: AU, UA, GC, CG
 - Wobble: GU, UG
- zwischen miteinander paarenden Basen müssen mindestens 3 Basen stehen
 $if(i, j) \in B \rightarrow i < j - 3$
 Beispiel Paarung A und U:

A	U	<u>A</u>	U	A	U	A	<u>U</u>
			1	2	3	4	
- keine Triplets (Multiplets): eine Base paart maximal mit einer anderen
 $if(i, j); (i, k) \in B \rightarrow j = k$
- keine pseudo-Knoten: Basen kreuzen sich nicht $if(i, j); (k, l) \in B \rightarrow i < j < k < l$ und $i < k < l < j$

Motivation zu Regeln: jedes Basenpaar teilt das Molekül in 2 Teile (innen und außen), die miteinander nicht interagieren (vor allem Regel 3 + 4)

physikalische Eigenschaften:

1. Großteil des stabilisierenden Energie für RNA-Struktur kommt aus der Sekundärstruktur
2. Sekundärstruktur bildet sich zeitlich vor Tertiärstruktur aus

Experimenteller Nachweis 3D, 4D:

- Röntgenkristallographie: Kristall benötigt → oft schwierig
- nuclear magnet resonanz (nmr): stark konzentrierte Lösung benötigt, nur Distanzen zwischen Atomen ermittelbar

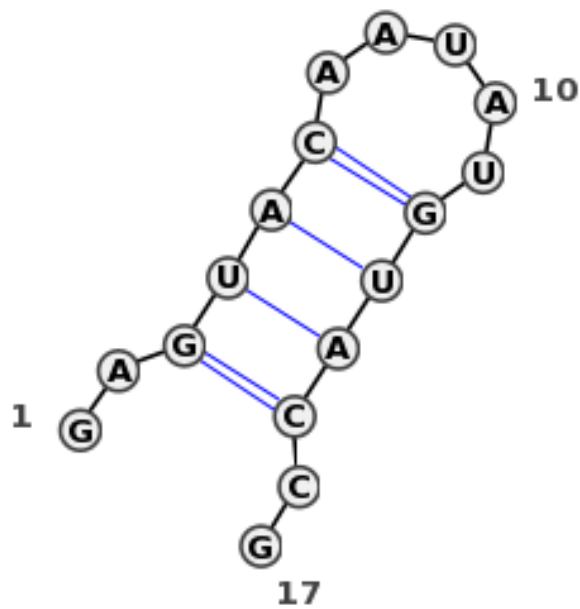
für 2D: Methoden, die bevorzugt einzelsträngige oder doppelsträngige Strukturen schneiden

2.3 Strukturabbildungen

1. Strukturplot:

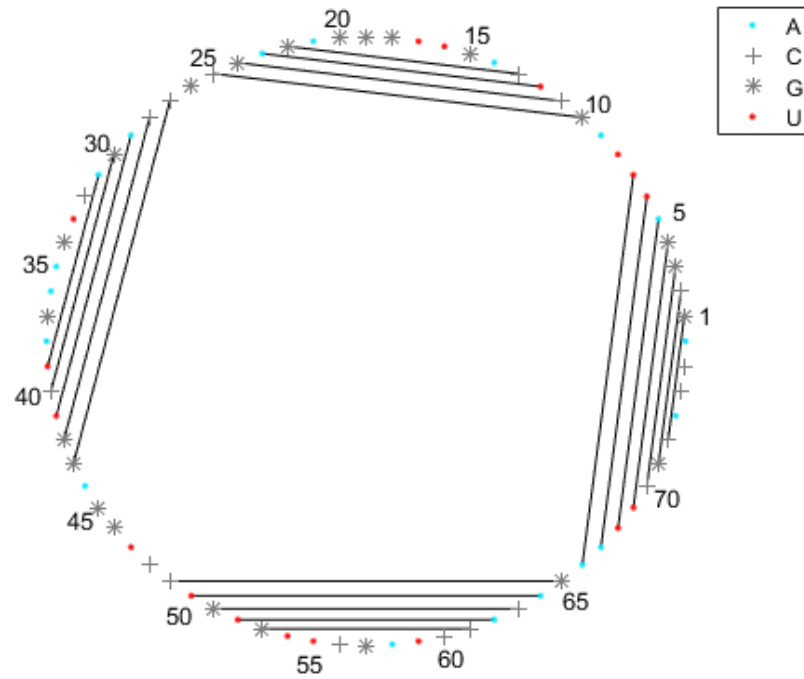


2. Dot-Bracket:

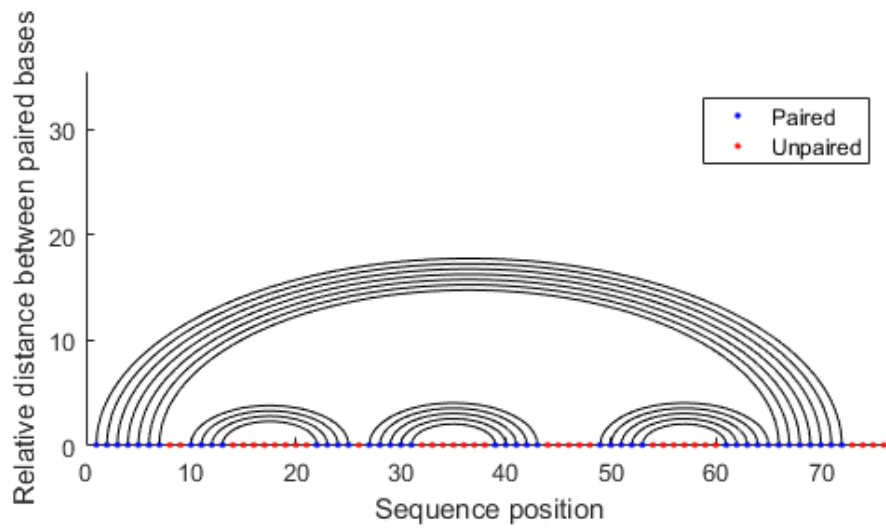


Seq:	GAGUACAAUAUGUACCG
Str:	..(((.....)))...

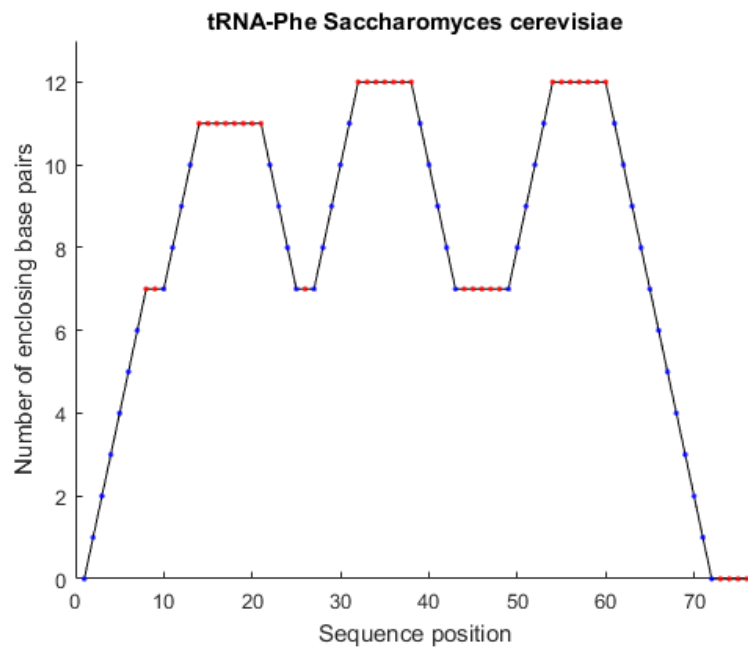
3. Zirkulärplot:



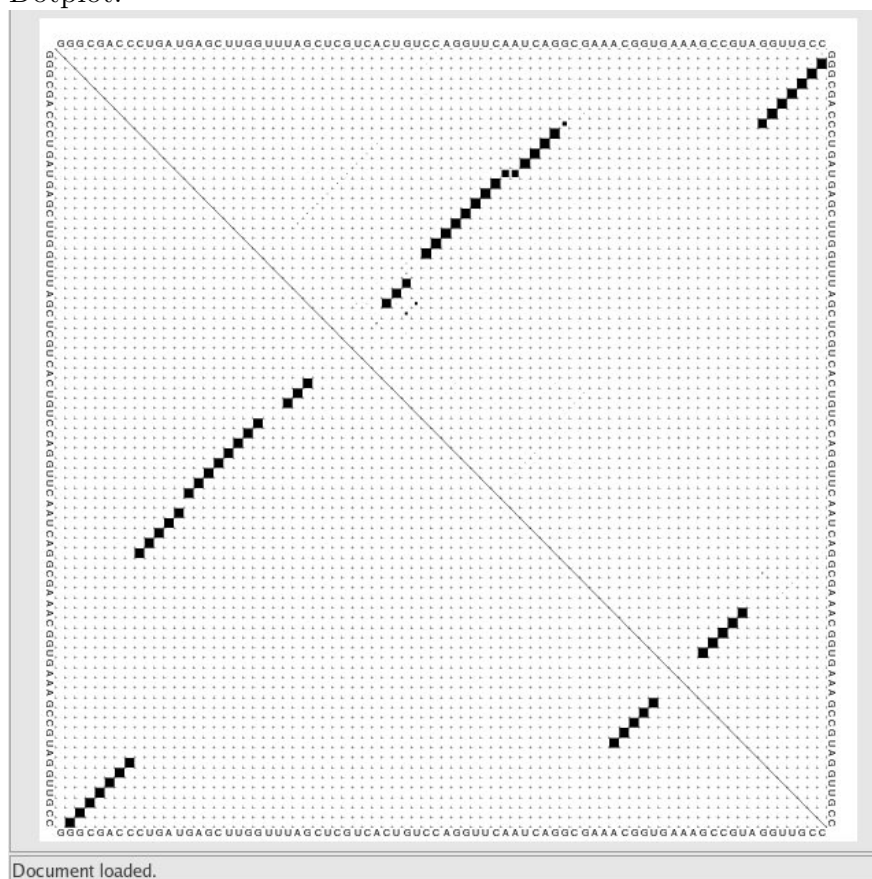
4. Bogenplot:



5. Mountainplot:



6. Dotplot:



3 Strukturvorhersage

- durch Aufteilung kann Dynamics Programming verwendet werden
- Beginn: einzelne Basen \rightarrow keine Struktur

3.1 Nussinov

- von Ruth Nussinov (1978)
- Versuch Struktur mit der maximalen Anzahl der Basenpaare zu finden (Grundlage ist Sequenz)

Dynamics Programming

- **Initialisierung:**

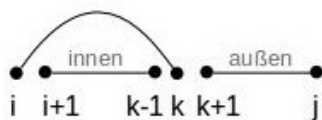
- $N(i, i) = 0$
- $N(i, j) = 0$ if $i < j \leq i + 3$ (siehe Regel 2)
- $N(j + 1, j) = 0$

- **Berechnung:**

$$N(i, j) = \max \begin{cases} N(i + 1, j) & (\text{ungepaart}) \\ \max_{i+3 < k \leq j} N(i + 1, k - 1) + N(k + 1, j) + F(i, k) \end{cases}$$

$$\text{mit } F(i, k) = \begin{cases} 1 & \text{if } i, k \in \{AU, GC, GU\} \\ -\infty & \text{else} \end{cases}$$

Basenpaarung mit i und k teilt Sequenz in inneren und äußeren Teil:



\rightarrow höchste Punktzahl wahrscheinlichste Sekundärstruktur

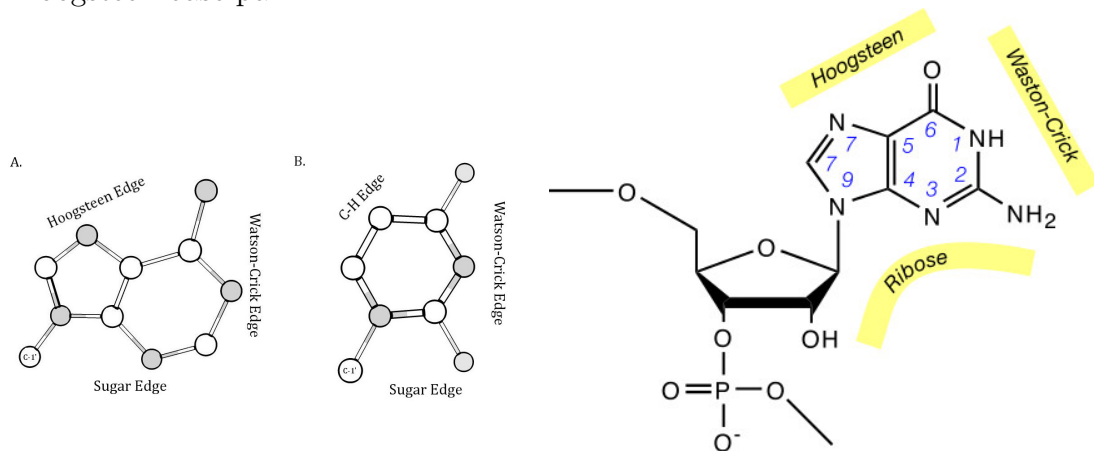
Ressourcenbedarf:

- Speicher: $\mathcal{O}(n^2)$
- Prozessor: $\mathcal{O}(n^3)$

- 3.2 Turner-Modell (Nearest-Neighbor-Modell)**
- 3.3 Zuker-Algorithmus**
 - 3.3.1 suboptimales Falten**
- 3.4 Wuchty-Algorithmus**
 - 3.4.1 Wuchty-Backtracking**
- 3.5 McCaskill**
- 3.6 stochastisches Backtracking**
- 3.7 Strukturvorhersagen verbessern**
- 3.8 Konsensusstrukturvorhersagen**
- 3.9 Wie kann RNA evolvieren?**
 - 3.9.1 Neutrale Netzwerke**
 - 3.9.2 SHAPE-Abstraktion**
 - 3.9.3 Energielandschaften**
 - 3.9.4 Faltungskinetik**
 - 3.9.5 Barriers Trees**
 - 3.9.6 Flooding-Algorithmus**
 - 3.9.7 Co-transcriptional folding**

4 weitere Bindungsarten, erlaubte Basenpaare

- Hoogsteen base pair⁵



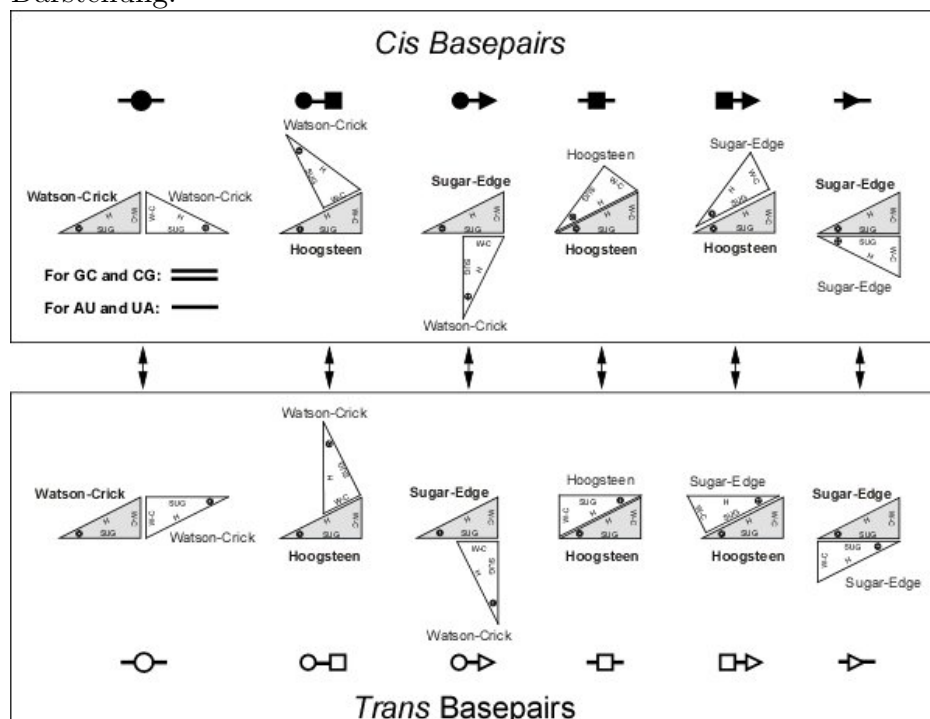
Jede Base kann mit jeder ihrer Kanten zu jeder Kante jeder Base ein Basenpaar bilden.

Non-Standard Basepairs:

Strukturmotive: Pattern von Standard basepairs führt zu speziellen 3D-Struktur (Kink-Turn)

Bifurcations (tripletts meistens) $12 * 12 * 2$ mögliche Basenpaare: Warum 288?

Darstellung:

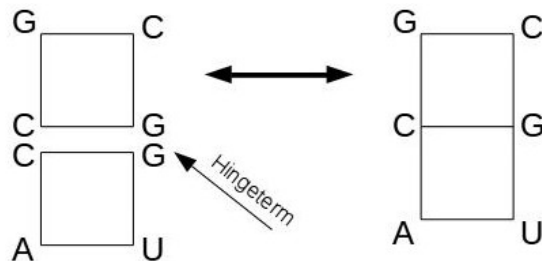


⁵https://en.wikipedia.org/wiki/Hoogsteen_base_pair

Isoelektrische Basenpaare

Änderung eines isoelektrischen Basenpaars gegen ein anderes ändert nichts an der Struktur

Listen von isoelektrischen Basenpaaren erstellt von Leontis und Westhof



Programme: MC-Fold, RNAWolf

5 Proteine

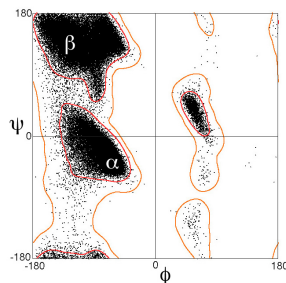
- 20 Aminosäuren
- drei positiv geladene Aminosäuren (basisch): Arg (R), His (H), Lys (K)
- zwei negativ geladene Aminosäuren (sauer): Asp (D), Glutaminsäure (E)
- sehr unterschiedlich in den Seitenketten
- Verbindung durch Peptidbindung

Frage: Wie rotieren Aminosäuren, die durch eine Peptidbindung verbunden sind, im Raum?

Stichworte: Cis, Torsionswinkel

Ramachandran Plot:⁶

allgemeines Beispiel:



⁶https://en.wikipedia.org/wiki/Ramachandran_plot

6 Sekundärstrukturelemente

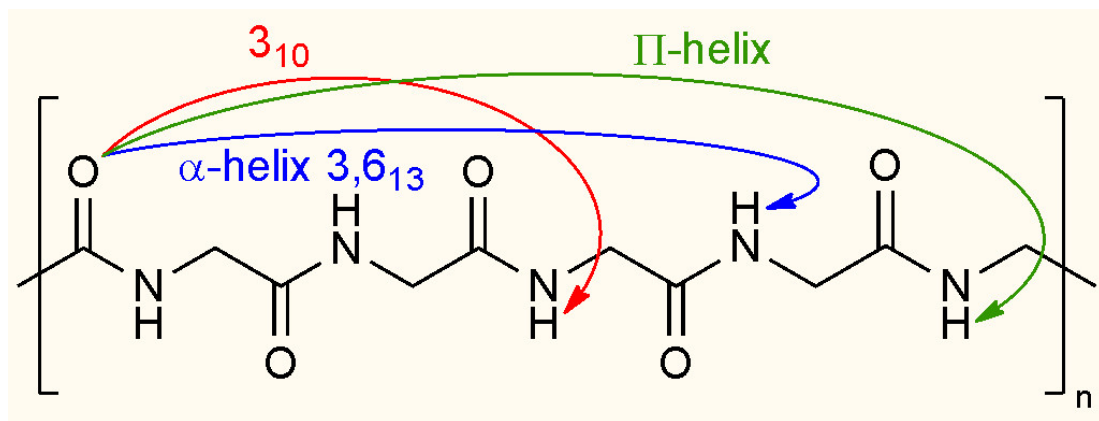
- Unterscheidung in drei Haupttypen⁷

Proteine:

- Helix α -Helix (häufigstes)
 - coiled-coil-Struktur: Helix umgeben mit einer Helix
 - Transmembranhelices: 20 - 30 Aminosäuren, hydrophob, gehen durch die Zellmembran durch
- Extended-Faltblatt: mindestens zwei Faltblätter immer zusammen, da diese sich gegenseitig stabilisieren
 - parallel, antiparallel
- Turn (drehen der Backbonerichtung)
- Coil (Rest)

drei Helixe: Unterscheidung, was und wie viel zwischen den Wasserstoffbrückenbindungen steht⁸

- α – Helix: 3,6,13-Helix (Helix zwischen 3. und 6. Atom, dazwischen liegen 13 Atome)
- π – Helix: 4,1,16



6.1 Chou-Fasman (Sekundärstrukturvorhersage von Proteinen)

- ca. 50% Genauigkeit

⁷<https://de.wikipedia.org/wiki/Sekund%C3%A4rstruktur>

⁸https://en.wikipedia.org/wiki/Protein_secondary_structure

- 3 Tabellen mit Scores für α (Helix), β (Faltblatt) und t (Turn) für alle Aminosäuren
 - z.B. gut für Helix: Glu (1,51), Met Ala, Leu
 - schlecht für Helix: Pro, Gly (0,57)
 - gut für Faltblatt: Val (1,7), Ile (1,6)
 - schlecht für Faltblatt: Asp, Glu (0,37), Pro (0,55)
- Unabhängig voneinander α, β, t bewerten:
 - nucleation: 4 von 6 Aminosäuren haben $S_{(\alpha)} \geq 1,03$
Erweitern nach links und rechts, bis Durchschnitt der letzten 4 AS $S_{(\alpha)} \geq 1$ haben
 - β : 3 von 5 Aminosäuren sollen $S_{(\beta)} \geq 1$ haben, letzten 4 AS $S_{(\beta)} \geq 1$
- Turn: $score(t) = S_{(t)}(x1) \cdot S_{(t)}(x2) \cdot S_{(t)}(x3) \cdot S_{(t)}(x4)$

Weiterentwicklung:

- nicht nur eine Aminosäure sondern gesamte Umgebung anschauen

GOR-Algorithmus:⁹

- bis zu 70% genau - es gibt GOR1 bis GOR5, unterschiedliche Berechnungen

- drei Matritzen mit Scores
20 x 17 Matritze ($\alpha, \beta, turn$)
Beispiel für α : waagrecht: -8 bis +8, senkrecht alle Aminosäuren
- Score aus Summierung über Matrixeinträge, dann ähnliche wie Chou-Fasman

Beispiel: ACCTYRARRRGHSTFYSW

für R $S_{\alpha} = S^{\alpha}(-8, A) + S^{\alpha}(-7, C) + \dots + S^{\alpha}(8, W)$

- das für alle Sekundärstrukturelemente

weiterer Algorithmus: SPIDER2

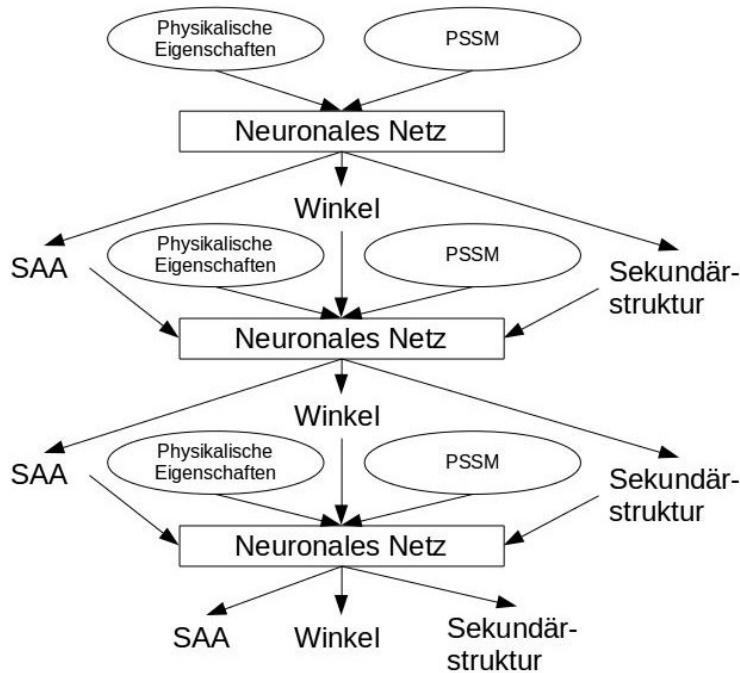
- ca. 80% genau
- Winkel zwischen Aminosäuren berechnen
- Surface Accesible Area
- Sekundärstrukturen

Physikalische Eigenschaften von Aminosäuren:

⁹https://en.wikipedia.org/wiki/GOR_method

- sterischer Parameter (graph shape index: dünnes oder dickes Molekül)
- Hydrophobizität
- Polarisierbarkeit
- Isoelektrischen Punkt
- Helix Wahrscheinlichkeit
- Volumen
- Falblattwahrscheinlichkeit
- zusätzlich mit psi-Blast: PSSM ermitteln (kein Ergebnis für Struktur sondern nur für Sequenz!)

dann alle diese Parameter in neuronales Netz stecken:



weitere Möglichkeit: Meta Server

- ruft mehrere Algorithmen auf
- höhere Wahrscheinlichkeit durch vergleichen der Ergebnisse (z.B. majority vote)

7 (Protein-) Strukturvorhersage (3D)

7.1 Strukturaufklärung

- Röntgen-Kristallographie
- NMR

7.2 Qualität der Strukturvorhersage

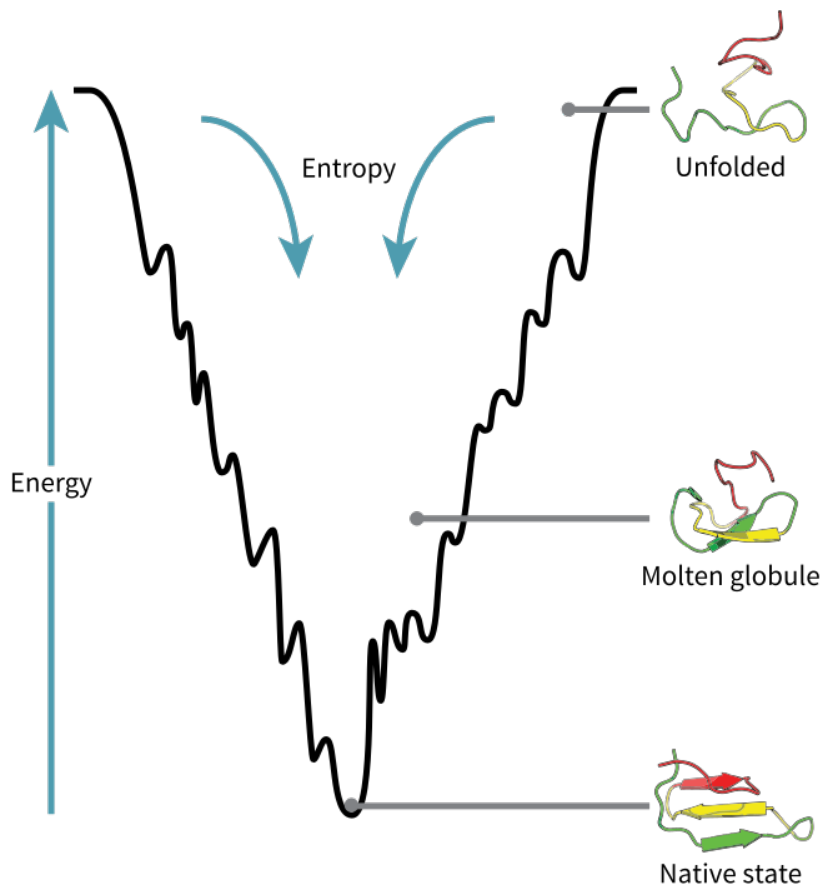
- RMSD (root mean square deviation): mittlerer Abstand in Å (10^{-10} m)

1-2 Å RMSD ist ein (sehr) guter Wert.

7.3 Problem der Strukturvorhersage (Levinthal-Paradoxon)

Eine Polypeptidkette von 100 Residuen hat 99 Peptidbindungen und daher 198 verschiedene Φ - und Ψ -Winkel. Wenn nun jeder dieser Winkel drei stabile Konformationen einnehmen kann, so ergibt sich eine Anzahl verschiedener Proteinstrukturen von 3^{198} . Wenn die Peptidkette bei der Faltung zum Protein nacheinander jeden dieser Winkel ausprobieren würde, bräuchte es länger als der Alter des Universums, um korrekt zu Falten.¹⁰

¹⁰https://en.wikipedia.org/wiki/Levinthal_paradox/



In der Natur falten Proteine aber im Bereich von Millisekunden. Wie ist das zu erklären?

Lokale Interaktionen führen den Faltungsprozess und schränken die Möglichkeiten ein. Experimente zeigen die resultierenden Intermediates und Transition states. Struktur und Faltung sind also *sequenzkodiert*.

7.4 Protein-Domains (Domänen)

- Protein-Untereinheit
- Falten unabhängig vom Rest des Proteinstrukturen
- Meistens funktionelle Untereinheit
- ca. 2700 Familien
- ca. 120000 Proteine (pdb)
- ca. 2/3 sind Multidomain-Proteine
- ca. 1224 Folds
 - Folds (SCOPE-Datenbank) Structural classification of proteins

- All α
- All β
- α/β , abwechselnde $\alpha/\beta \Rightarrow$ parallele β -Faltblätter
- $\alpha + \beta$, getrennte $\alpha, \beta \Rightarrow$ antiparallele β -Faltblätter
- Multidomain ($\alpha + \beta$)
- Andere (coiled coil, membrane, cell-surface)

7.5 Zwei Typen von Vorhersagen

- Ab initio
- Template based
 - Homology based
 - Threading

7.5.1 Ab-initio-Vorhersage

- Suche Strukturvorschläge
- Bewerten der Strukturen
 - physikalisch
 - knowledge-based $\log(\frac{\text{observed}}{\text{expected}})$

Physikalisch Molecular force field

E_{Bindung}

- Bindungen
 - Abstand
 - Winkel α (Bindung)
- Winkel ϕ (Torsion)

$E_{\text{ungebunden}}$

- Ladungen
- Dipol

$$E = E_{\text{Bindung}} + E_{\text{Nicht-Bindung}}$$

$$E_{\text{Bindung}} = \sum_{\alpha} k(\alpha - \alpha_0)^2$$

$$+ \sum_{\text{Bindungen}} k(r - r_0)^2$$

$$+ \sum_{\phi(\text{Torsion})} \frac{V_n}{2}(1 - \cos(n\phi - \gamma))$$

$$\text{Alternative für Bindungspotential (Morse-Potential¹¹) } \sum_{\text{Bindungen}} D_e * (1 - e^{-a(r-r_0)^2})$$

k Kraftkonstante

α Bindungswinkel

r Bindungslänge

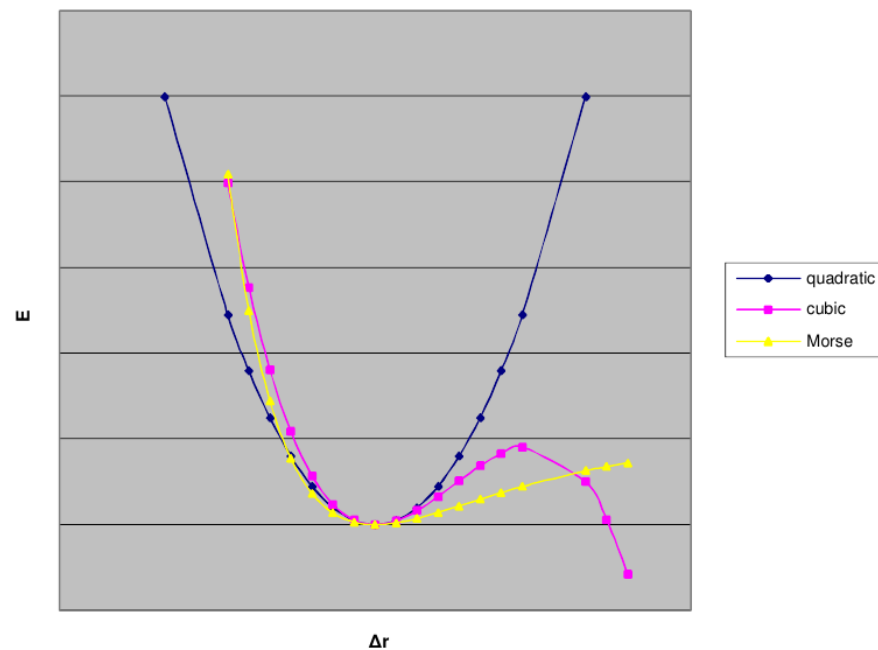
r_0 Bindungslänge mit der geringsten potentiellen Energie

D_e Dissoziationsenergie

$a = (0.5 * \frac{k}{D_e})^{1/2}$ "Steifigkeits-" Konstante

V_n Barrier height

γ Phasenverschiebung



$$E_{\text{Nicht-Bindung}} =$$

$$\sum_{i,j \in \text{Atome}} \frac{P_i P_j}{\epsilon r_{i,j}} \text{ Ladung: Coulomb-Terme}$$

$$+ \sum_{\text{Paar}} \frac{c}{r^{12}} - \frac{c}{r^6} \text{ Dipol: Van-der-Waals-Kräfte, Lennard-Jones-Potential 12, 6}$$

cut-off-radius

¹¹https://en.wikipedia.org/wiki/Morse_potential

Lösungsmittel

- implicit solvent
- explicit solvent

Spezialterme: H-Terme, Π -Interaktionen

Knowledge based

- coarse-graining
- c_α als Beschreibung der AS
- Alle Backbone-Atome
- Alle Backbone-Atome + repräsentativ die Sk (center of mass)
- Alle Atome

Einfache Potentiale

- Abstand der Aminosäuren
- Nachbarschaft der Aminosäuren

QUARK

- Backbone atomweises Paar-Potential
- Sk-Schwerpunkt
- Excluded volume
- H-Bindungen
- Surface accessible area
- Torsionswinkel im Backbone
- Distanzen von Fragmenten
- Gyrationradius
- $\beta\alpha\beta$ -linkshändig (HIER FEHLT NOCH WAT)
- $\beta\alpha\beta$ -packing
- α -packing
- β -packing

Konformation \Rightarrow dafür Minimum

- Steepest descent
- Conjugate gradient
- Newton-Verfahren
- Monte-Carlo-Verfahren
- Simulated annealing

7.5.2 Template based methods

Homology based

- Sequenzalignment zu den Sequenzen der bekannten Strukturen
- Alignment der Sequenz zur Struktur des Kandidaten
- Bauen einer Struktur aus dem Alignment
- Bewerten der Struktur

HIER FEHLT EIN BILDCHEN

- Verbinden der Distanzpotentiale (gewichtet, multiplikativ)

CASP (critical assessment of structure prediction)

Threading

1. Sequenz-Struktur-Alignment zur Identifizierung der Kandidaten

$$\begin{pmatrix} \alpha \\ 0.5 \\ P \\ \text{groß} \end{pmatrix} \begin{pmatrix} \alpha \\ 0.3 \\ P \\ \text{klein} \end{pmatrix}$$

2. Bauen einer Struktur aus Alignment
3. Bewerten der Struktur