Rekonstruktion phylogenetischer Bäume – ein Maximum Likelihood Ansatz

Florian Sohler

10. Juni 1998

1 Einführung

Wir haben verschiedene Methoden kennengelernt, aus DNA- oder Protein-Sequenzen phylogenetische Bäume zu rekonstruieren. Den aus der Statistik motivierten Maximum Likelihood Ansatz werde ich nun genauer analysieren und einen konkreten Algorithmus vorstellen, der mit diesem Ansatz arbeitet. Wir werden hierbei mehrfach auf die Verwendung von Heuristiken stoßen, da die Berechnungskomplexität bisher entwickelter Algorithmen ohne diese Heuristiken zu hoch ist. Ich gehe im folgenden immer davon aus, daß eine Reihe alignierter DNS-Sequenzen homologer Gene, also Gene, die einen gemeinsamen Vorfahren haben, verschiedener Spezies vorliegt, deren phylogenetischer Baum gesucht ist.

2 Maximum Likelihood Methoden

Maximum Likelihood ist ein Begriff aus der Statistik, genauer aus der Schätztheorie. Zu dem Ergebnis eines Zufallsexperiments wird ein Parameter der zugehörigen Wahrscheinlichkeitsverteilung geschätzt.

Definition. Gegeben sei eine Familie von Wahrscheinlichkeitsverteilungen $\{P_\theta : \theta \in \Theta\}$. Eine Funktion L_x, die jedem Wert θ die Wahrscheinlichkeit von x unter der durch θ gegebenen Wahrscheinlichkeitsverteilung zuordnet, nennen wir eine **Likelihood-Funktion**. Also

$$L_x(\theta) = P_\theta(x)$$

Falls L_x maximal wird in $\hat{\theta}(x)$, also

$$L_x(\hat{\theta}(x)) = sup\{L_x(\theta) : \theta \in \Theta\},$$

nennen wir $\hat{\theta}(x)$ eine **Maximum-Likelihood-Schätzung** von θ.

In unserem Fall bezeichnet x die Menge der vorliegenden DNS-Sequenzen als Matrix geordnet (Abb. 1), x_i sei eine Spalte aus dieser Matrix, also eine Position in den Sequenzen. Θ steht für den Raum aller Topologien binärer Bäume mit den Sequenzen aus x als Blättern. Man beachte hierbei, daß die geschätzte Topologie keineswegs die wahrscheinlichste ist, da L_x nicht die Wahrscheinlichkeit dafür angibt, daß der Baum der richtige ist, L_x ist nicht einmal eine Wahrscheinlichkeitsverteilung.

$$\begin{vmatrix} A & A & C & T & G & G & T & C & A & A & C & C & A & C & T \\ A & A & A & G & G & C & T & C & A & A & C & A & A & C & T \\ A & A & A & G & G & C & T & A & A & A & C & C & A & C & T \\ G & A & C & T & G & T & T & A & A & G & C & C & A & G & T \\ G & C & C & T & G & T & T & A & C & G & C & A & C & G & T \end{vmatrix}$$

Abbildung 1: Matrix der Sequenzen

3 Berechnung der Likelihood Funktion

Um Maximum Likelihood Methoden verwenden zu können, müssen wir nun eine Berechnungsvorschrift für die Wahrscheinlichkeit der DNS-Sequenzen bei einem vorgegebenen phylogenetischen Baum finden. Gesucht ist also eine Funktion $L_x(\theta)$, die einer Baumtopologie θ die Wahrscheinlichkeit der zur Untersuchung vorliegenden DNS-Sequenzen x zuordnet. Dazu möchten wir ein Modell finden, das auf einfache Art und Weise die Evolution beschreibt. Dies scheint nicht möglich zu sein, da insbesondere der Prozeß der Auslöschung und Bildung neuer Spezies und der der Selektion nicht modellierbar zu sein scheint. Wir beschränken uns daher auf ein Modell für die Veränderung der DNS. Hierzu möchte ich ein einfaches Modell vorstellen, das die Änderung der Basen an einer Stelle der DNS als Markovprozeß beschreibt.

Eine wichtige Annahme ist die Unabhängigkeit der Veränderungen an verschiedenen Stellen der DNS. Dies ist eine starke Einschränkung, die unter anderem die Modellierung von Einfügungen und Löschungen von Basen unmöglich macht. Mit dieser Annahme können wir uns darauf beschränken, die Vorgänge an einer Stelle der DNS zu betrachten. Hierzu berechnen wir die Wahrscheinlichkeiten $P_{ij}(t)$, wobei die i, j Werte von 1 bis 4 annehmen, entsprechend den vier Basen A, C, G und T. $P_{ij}(t)$ gibt die Wahrscheinlichkeit an, daß eine Base i nach einer Zeit t durch die Base j ersetzt wird.

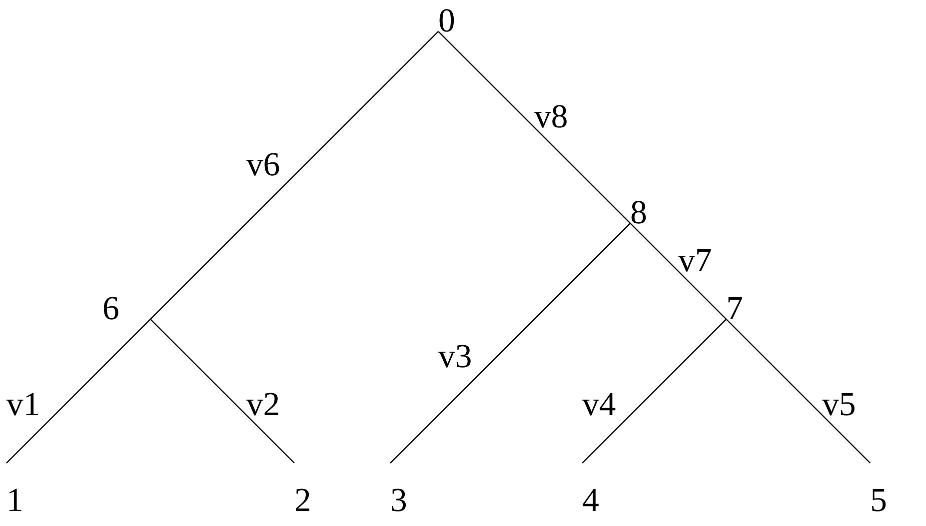

Abbildung 2: Ein Beispielbaum

Mit diesen Wahrscheinlichkeiten $P_{ij}(t)$ können wir dann den Likelihood-Wert des gesamten Baumes an einer Stelle der DNA berechnen. Ich möchte das nun am Beispiel von Abbildung 2 zeigen. Legen wir die Basen an allen Stellen des Baumes fest als s_0 bis s_8, so ergibt sich die Wahrscheinlichkeit für die beobachteten Blätter aus der Wahrscheinlichkeit für den Anfangszustand π_{s_0} und den Übergangswahrscheinlichkeiten $P_{ij}(v_j)$, wobei die v_j die verstrichene Zeit zwischen zwei Knoten angeben.

$$L = \pi_{s_0} P_{s_0 s_6}(v_6) P_{s_6 s_1}(v_1) P_{s_6 s_2}(v_2) P_{s_0 s_8}(v_8) P_{s_8 s_3}(v_3) P_{s_8 s_7}(v_7) P_{s_7 s_4}(v_4) P_{s_7 s_5}(v_5) \tag{1}$$

Da die Basen in den inneren Knoten jedoch nicht festliegen, müssen wir über alle möglichen Belegungen summieren. Definieren wir $L_s(k)$ als den Likelihood-Wert am Knoten k, falls s der Zustand dort ist, kann man für den Knoten k, dessen Nachfolger i und j seien, folgende Rekursionsformel angeben, wobei b_k die tatsächlich beobachtete Base am Blatt b bezeichnet:

$$L_{s_k}(k) = \begin{cases} \delta_{s_k b_k} & \text{falls } k \text{ ein Blatt ist} \\ (\sum_{s_i} P_{s_k s_i(v_i)} L_{s_i}(i))(\sum_{s_j} P_{s_k s_j(v_j)} L_{s_j}(j)) & \text{sonst} \end{cases} \tag{2}$$

Damit ergibt sich für den gesamten Baum θ mit Wurzel 0 folgende Formel:

$$L_{x_i}(\theta) = \sum_{s_0} \pi_{s_0} L_{s_0}(0) \tag{3}$$

Nun haben wir den Likelihood-Wert für eine Stelle in den zu untersuchenden Sequenzen bestimmt. Das Produkt aller dieser Werte bildet dann den Likelihood-Wert für die volle Länge der Sequenzen.

Jetzt fehlen uns nur noch die $P_{ij}(t)$, um die Likelihoodberechung durchführen zu können. Wir haben bei deren Definition implizit vorausgesetzt, daß sie unabhängig vom Zustand zu früheren Zeitpunkten sind. Ein solches System beschreibt man gewöhnlich als umkehrbaren zeithomogenen Markovprozeß:

3.1 Markovprozesse

Definition. Ein stochastischer Prozeß hat die **Markoveigenschaft**, wenn $Prob(X(t+s) = j|X(t) = i)$ nur abhängt von s, i, j und t. Wenn diese Wahrscheinlichkeit auch von t unabhängig ist, so spricht man von einem **zeithomogenen Markovprozeß**. Dann können wir die **Übergangsmatrix** P definieren durch $P_{ij}(s) = Prob(X(t+s) = j|X(t) = i)$.

Für die Übergangsmatrix eines zeithomogenen Markovprozesses gilt die Chapman-Kolmogorov-Gleichung:

$$P(s+t) = P(s)P(t) \text{ und insbesondere} \tag{4}$$
$$P(0) = I \tag{5}$$

Hieraus folgen nach kurzer Rechnung die Kolmogorov-Differentialgleichungen (falls die entsprechenden Grenzwerte existieren):

$$P'(t_0) = P(t_0)P'(0) = P(t_0)Q \tag{6}$$

$Q = P'(0)$ nennt man auch die **Ratenmatrix** des Markovprozesses, denn für kleine δt gilt:

$$P(\delta t) = \delta t Q$$

Gleichung 6 wird gelöst durch

$$P(t) = e^{Qt} = \sum_{k=0}^{\infty} t^k Q^k / k!. \tag{7}$$

Für die Übergangsmatrix $P(t)$ gilt $\sum_k P_{ik}(t) = 1$, da jede Spalte einen Wahrscheinlichkeitsvektor darstellt. Daher folgt für Q:

$$\sum_k Q_{ik} = \sum_k \frac{d}{dt} P_{ik}(t)|_{t=0} = \frac{d}{dt} \sum_k P_{ik}(t)|_{t=0} = \frac{d}{dt} 1|_{t=0} = 0$$

Ein Wahrscheinlichkeitsvektor π, für den gilt $\pi^T Q = \pi^T$, nennt man Gleichgewichtsvektor. Eine stärkere Eigenschaft ist das detaillierte Gleichgewicht, welches für den Gleichgewichtvektor π fordert, daß $\pi_i P_{ij}(t) = \pi_j P_{ji}(t)$. Gleichwertig dazu ist die entsprechende Bedingung für die Ratenmatrix: $\pi_i Q_{ij} = \pi_j Q_{ji}$. Mit dem detaillierten Gleichgewicht ist die Zeitumkehrbarkeit des Markovprozesses gegeben.

3.2 Austauschmodelle

Für die Modellierung des Basenaustausches in der DNS werden nur umkehrbare Markovprozesse benutzt. Man geht davon aus, daß die Verteilung der Basen den Gleichgewichtszustand erreicht hat, d.h. die Wahrscheinlichkeit, an einer beliebigen Position der DNS die Base i vorzufinden entspricht der Wahrscheinlichkeit π_i, wobei π der Gleichgewichtsvektor des Markovprozesses ist. Man benutzt diese Eigenschaft so, daß man die relativen Häufigkeiten der Basen in den Sequenzen bestimmt und diese als Gleichgewichtsvektor π annimmt. Dann berechnet man die Einträge der Ratenmatrix, die allgemein folgende Form hat:

$$Q = \begin{pmatrix} -\mu(a\pi_C + b\pi_G + c\pi_T) & \mu a\pi_C & \mu b\pi_G & \mu c\pi_T \\ \mu a\pi_A & -\mu(a\pi_A + d\pi_G + e\pi_T) & \mu d\pi_G & \mu e\pi_T \\ \mu b\pi_A & \mu d\pi_C & -\mu(b\pi_A + d\pi_C + f\pi_T) & \mu f\pi_T \\ \mu c\pi_A & \mu e\pi_C & \mu f\pi_G & -\mu(c\pi_A + e\pi_C + f\pi_G) \end{pmatrix} \tag{8}$$

Im einfachsten Fall geht man davon aus, daß $a = b = c = d = e = f = 1$ ist und die relative Häufigkeit der aller Basen $\pi_i = \frac{1}{4}$. Dann erhält man das Jukes-Cantor-Modell:

$$Q = \begin{pmatrix} -\frac{3}{4}\mu & \frac{1}{4}\mu & \frac{1}{4}\mu & \frac{1}{4}\mu \\ \frac{1}{4}\mu & -\frac{3}{4}\mu & \frac{1}{4}\mu & \frac{1}{4}\mu \\ \frac{1}{4}\mu & \frac{1}{4}\mu & -\frac{3}{4}\mu & \frac{1}{4}\mu \\ \frac{1}{4}\mu & \frac{1}{4}\mu & \frac{1}{4}\mu & -\frac{3}{4}\mu \end{pmatrix} \tag{9}$$

Meistens werden jedoch zumindest unterschiedliche Wahrscheinlichkeiten für Transitionen und Transversionen angenommen. Transversionen sind Wechsel zwischen Purinen und Pyrimidinen, Transitionen sind Wechsel innerhalb einer der beiden Gruppen.

3.3 Folgerungen

Aus der Umkehrbarkeit des Markovprozesses kann man folgern, daß die Längen der beiden Kanten an der Wurzel des Baumes nur durch ihre Summe in die Berechnung der Likelihood-Funktion eingehen. Man muß also anstelle unseres Beispielbaumes aus Abbildung 2 den ungewurzelten Baum, der in Abbildung 3 gezeigt ist, betrachten, denn nur dieser wird letztendlich geschätzt, da das Einfügen einer Wurzel an einer beliebigen Stelle den Likelihood-Wert nicht verändern würde. Der ungewurzelte Baum entspricht also einer ganzen Klasse von gewurzelten Bäumen, nämlich denen, die durch das Einfügen einer Wurzel erzeugt werden können.

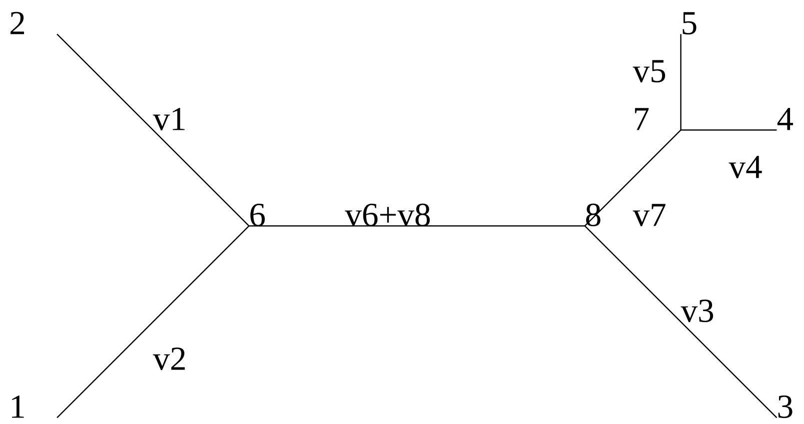

Abbildung 3: Der ungewurzelte Beispielbaum

Damit sind wir nun in der Lage, zu der Topologie eines ungewurzelten, binären Baumes den Likelihood-Wert zu berechnen. Wir nehmen eine Wurzel an einer beliebigen Stelle an und wenden Gleichung 3 für jede Stelle der Sequenzen an und bilden das Produkt über all diese Werte:

$$L_x(\theta) = \prod_i L_{x_i}(\theta) \tag{10}$$

4 Maximierung der Likelihood Funktion

Um die gesuchte Maximum-Likelihood-Schätzung $\hat{\theta}$ zu erhalten, muß nun die im vorigen Abschnitt berechnete Likelihood-Funktion über alle möglichen Topologien und Kantenlängen maximiert werden. Zunächst optimieren wir für eine feste Topologie die Längen der Kanten. Joseph Felsenstein zeigt hierzu in [Fels81] ein iteratives Verfahren, das immer eine Länge optimiert, während die anderen festgehalten werden, bis auf diese Weise keine weitere Verbesserung mehr möglich ist. Die Suche nach der besten Topologie stellt uns vor ein großes Problem. Die einfache analytische Lösung bleibt uns versperrt, da auf dem Raum der Baumtopologien keine Differentiation möglich ist. Die erschöpfende Suche über alle möglichen Topologien ist ebenfalls keine Lösung, da die Anzahl der Topologien zu groß ist. Felsenstein schlägt einen Greedy-Algorithmus vor, der von der Reihenfolge eingegebenen Sequenzen abhängt. Die Sequenzen werden nacheinander eingefügt. Für jede Sequenz wird jede Kante als möglicher Einfügeort getestet. Die Kante, die zum höchsten Likelihood-Wert führt, wird genommen. Bevor mit der nächsten Sequenz fortgefahren wird, werden lokale Veränderungen am Baum vorgenommen, bis auch diese keine Verbesserung mehr bringen. Man kann dieses Verfahren nun mehrfach mit unterschiedlicher Eingabereihenfolge durchführen und die möglicherweise unterschiedlichen Ergebnisse auf verschiedene Weise verbinden oder einfach das mit dem höchsten Likelihood-Wert als Gesamtergebnis akzeptieren.

Im nächsten Abschnitt stelle ich einen Ansatz vor, der auf die Berechnung von Likelihood-Werten ganzer Bäume völlig verzichtet, das Quartet Puzzling.

5 Das PUZZLE-Programm

PUZZLE ist ein Programm, das Maximum Likelihood Methoden verwendet. Es werden jedoch nur Likelihood-Werte von kleinen Bäumen mit jeweils vier Blättern, sogenannten Quartetten, berechnet. Dann werden die Quartette unter Verwendung von Heuristiken zu einem Gesamtbaum zusammengesetzt.

Wenn wir n Sequenzen zu untersuchen haben, gibt es $\binom{n}{4}$ Quartette. Ein Quartett hat nur drei mögliche Topologien, so daß wir leicht die Topologie mit dem maximalen Likelihood-Wert bestimmen können. Diese “Maximum-Likelihood-Quartette”

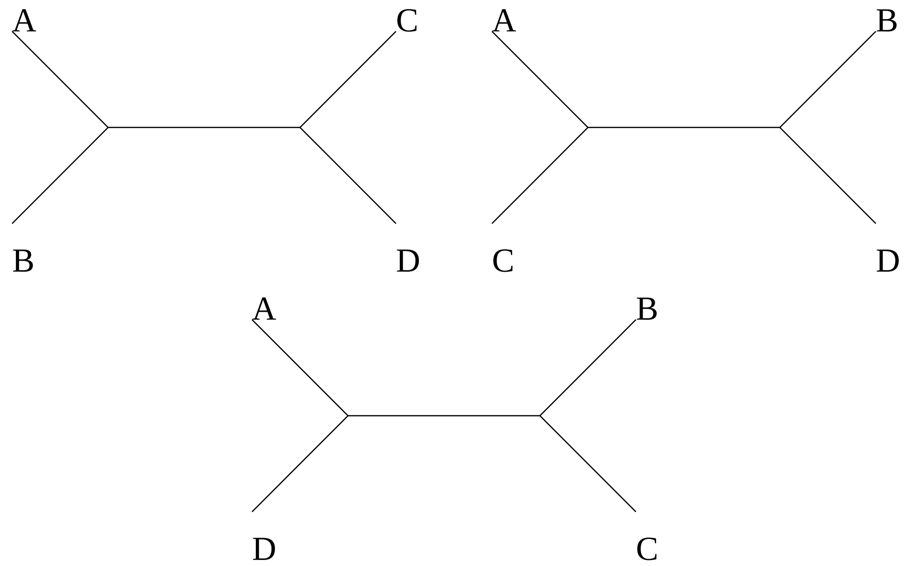

Abbildung 4: Die drei möglichen Topologien eines Quartetts

definieren eine Nachbarschaftsrelation, z.B. $AB||CD$ für die erste Topologie in Abbildung 4. Dies bedeutet, daß zumindest eine Kante die beiden Sequenzen A und B von C und D trennt, oder anders ausgedrückt, daß die Wege von A nach B und C nach D sich nicht überschneiden Mit Hilfe dieser Relation kann man einen Algorithmus definieren, der durch sequentielles Einfügen von Sequenzen einen Zwischenbaum erzeugt. Abbildung 5 zeigt diesen Einfügevorgang. Jedes Quartett, das die einzufügende Sequenz enthält, wird betrachtet. Wenn $ij||kl$ ein solches Quartett bezeichnet und l die einzufügende Sequenz ist, wird jeder Kante auf dem Weg zwischen i und j ein Strafpunkt gegeben. Nachdem dies für alle Quartette durchgeführt wurde, wird l bei der Kante mit der niedrigsten Punktzahl eingefügt. Auch diser Algorithmus ist abhängig von der Reihenfolge der Sequenzen. Er wird so oft wie möglich mit randomisierter Eingabereihenfolge durchgeführt, wobei alle Zwischenbäume gespeichert werden. Diese Zwischenbäume werden dann zu einem Konsensbaum zusammengefügt. Der Konsensbaum wird nach einer sehr einfachen Regel gebildet: Jede Kante, die in mehr als der Hälfte der Zwischenbäume vorkommt, wird in den Konsensbaum aufgenommen. Mit den folgenden zwei Definitionen zeigt man leicht, daß der konstruierte Baum wieder ein ungewurzelter Baum mit den Sequenzen als Blättern ist.

Definition. Ein **ungewurzelter Baum** T über der Blattmenge S ist eine Menge von echten Teilmengen $M \subset S$, für die gilt:

1. $\forall i \in S : \{i\} \in T$

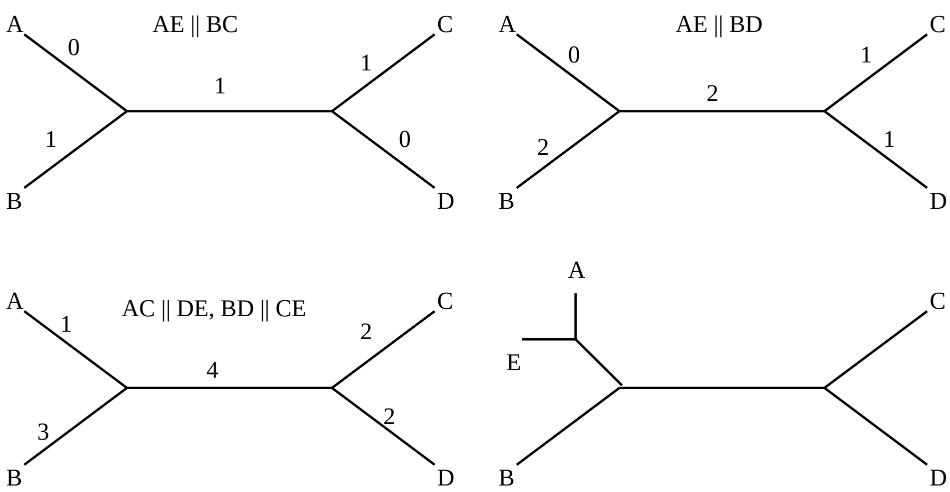

Abbildung 5: Einfügen der Sequenz E in einen vorhandenen Baum mit vier Sequenzen

2. $\forall A \in T : S \backslash A \in T$
3. $\forall A, B \in T$: Falls $A \cap B \neq \emptyset$, dann gilt $A \subset B$ oder $A \supset B$

Definition. Eine Menge $\hat{T}$ von Teilmengen von S heißt **Mehrheitskonsensbaum** zu einer endlichen Familie τ von ungewurzelten Bäumen über S, falls
$A \in \hat{T}$ genau dann, wenn $A \in T$ für mehr als die Hälfte aller $T \in \tau$.

Wenn man sich davon überzeugt hat, daß die obige Definition tatsächlich genau die ungewurzelten Bäume mit Blättern aus der Menge S beschreibt (man beachte dazu, daß eine der Teilmengen immer genau einem Unterbaum bzw. der darüberliegenden Kante entspricht), folgt leicht der

Satz. *Der Mehrheitskonsensbaum $\hat{T}$ zu τ ist ein ungewurzelter Baum über S.*

Beweis. Zu zeigen sind die drei Eigenschaften aus der Definition für ungewurzelte Bäume. 1. und 2. sind klar. Falls nun $A, B \in \hat{T}$, dann $\exists T \in \tau : A, B \in T$, da sowohl A als auch B in mehr als der Hälfte aller $T \in \tau$ sind. Damit folgt 3. □

Der Konsensbaum ist nun das Ergebnis des PUZZLE-Algorithmus. Dies ist nicht mehr unbedingt ein binärer Baum. Ein großer Vorteil des Algorithmus ist, daß man jeder Kante des Ergebnisbaumes einen Zuverlässigkeitswert zuordnen kann, nämlich den Prozentsatz, zu dem die Kante in den Zwischenbäumen vorhanden war.

6 Diskussion

Wie die meisten stochastischen Verfahren zeigt sich auch Maximum.Likelihood sehr robust gegenüber kleineren Schwankungen in den Daten und leichte änderungen im Modell. Dies ist natürlich ein großer Vorteil gegenüber Methoden, die hier empfindlich sind. Außerdem ermöglicht dieser Ansatz die Anwendung von Testverfahren der Statistik, auf die hier nicht näher eingegangen wurde. In der Praxis zeigt sich, daß Maximum-Likelihood-Methoden besonders dort besser funktionieren als zum Beispiel Parsimony-Methoden, wo große Änderungen in den Sequenzen stattgefunden haben, also die Verwandtschaft entfernter ist.

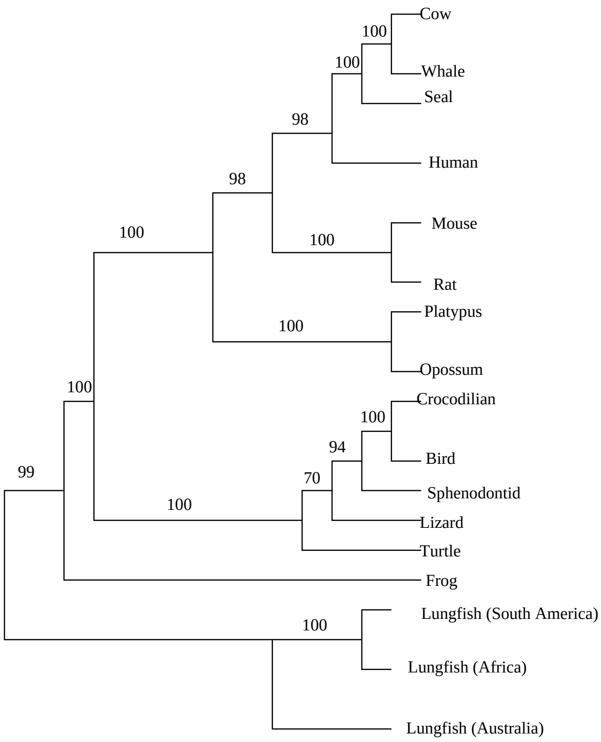

Abbildung 6: Ergebnis des PUZZLE-Programms am Beispiel der Amnioten

Leider ist der Aufwand zur Berechnung der Likelihoodfunktion relativ hoch, außerdem gibt es keinen Algorithmus, der mit akzeptablem Aufwand garantiert das globale Maximum über allen möglichen Topologien findet. Daher müssen Heuristiken entwickelt werden, wie am Beispiel des Quartet-Puzzling gezeigt, über die man jedoch meist keine theoretischen Aussagen treffen kann. Solche Heuristiken müssen daher durch Simulationen und Tests mit echten Daten auf ihre Qualität geprüft werden. Solche Tests und die tatsächliche Anwendung in der Praxis zeigen, daß Quartet ein schneller Algorithmus ist, der ausgezeichnete Ergebnisse liefert.

Literatur

[StrHae96] Korbinian Strimmer, Arndt van Haeseler: Quartet Puzzling: A Quartet Maximum-Likelihood Method for Reconstructing Tree Topologies, Mol. Biol. Evol. 13(7):964-969, 1996

[Fels81] Joseph Felsenstein: Evolutionary Trees from DNA Seqeunces: A Maximum Likelihood Approach, J. Mol. Evol. 17:368-376, 1981