Bioinformatika
Filogenetska stabla

Prof. G. Pavlović-Lažetić,
Matematički fakultet,
Beogradski univerzitet,
šk.2013/2014. g.
gordana@matf.bg.ac.rs

Stablo života

- Sličnost molekularnih mehanizama u organizmima sugeriše postojanje zajedničkog pretka svih živih bića na Zemlji
- Povezanost bilo kog skupa organizama naziva se *filogenijom*
- Ova povezanost se prikazuje *filogenetskim stablom*
- Zadatak filogenetike je da izvede to stablo iz opažanja nad postojećim organizmima
- Morfološke karakteristike tradicionalno se koriste za izvođenje filogenije
- Ako imamo skup sekvenci iz različitih vrsta, možemo da ih iskoristimo za izvođenje verovatne filogenije tih vrsta
- Pretpostavka je da su sekvence proistekle iz zajedničkog predačkog gena zajedničke predačke vrste

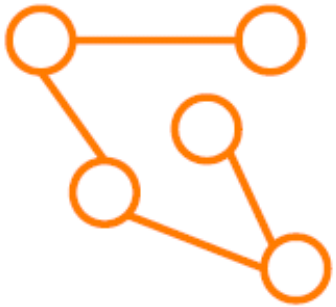
Filogenetsko izvođenje: zadatak

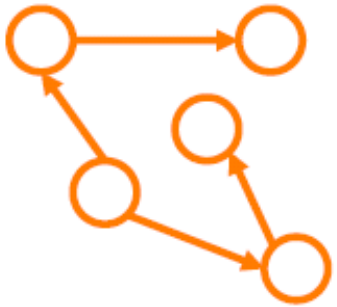
- Kada su *dati* podaci koji karakterišu vrste / gene

- *Izvesti* filogenetsko stablo koje ispravno karakteriše evoluciono poreklo među tim vrstama / genima

Stablo u kontekstu filogenetike

- U teoriji grafova:
  - Neusmeren slučaj: graf bez ciklusa
  - Usmeren slučaj: pripadni neusmereni graf je stablo
  - Ulazni red čvora ≤ 1
  - Stablo može da bude sa korenom (“koreno”) ili bez njega (“nekoreno”)





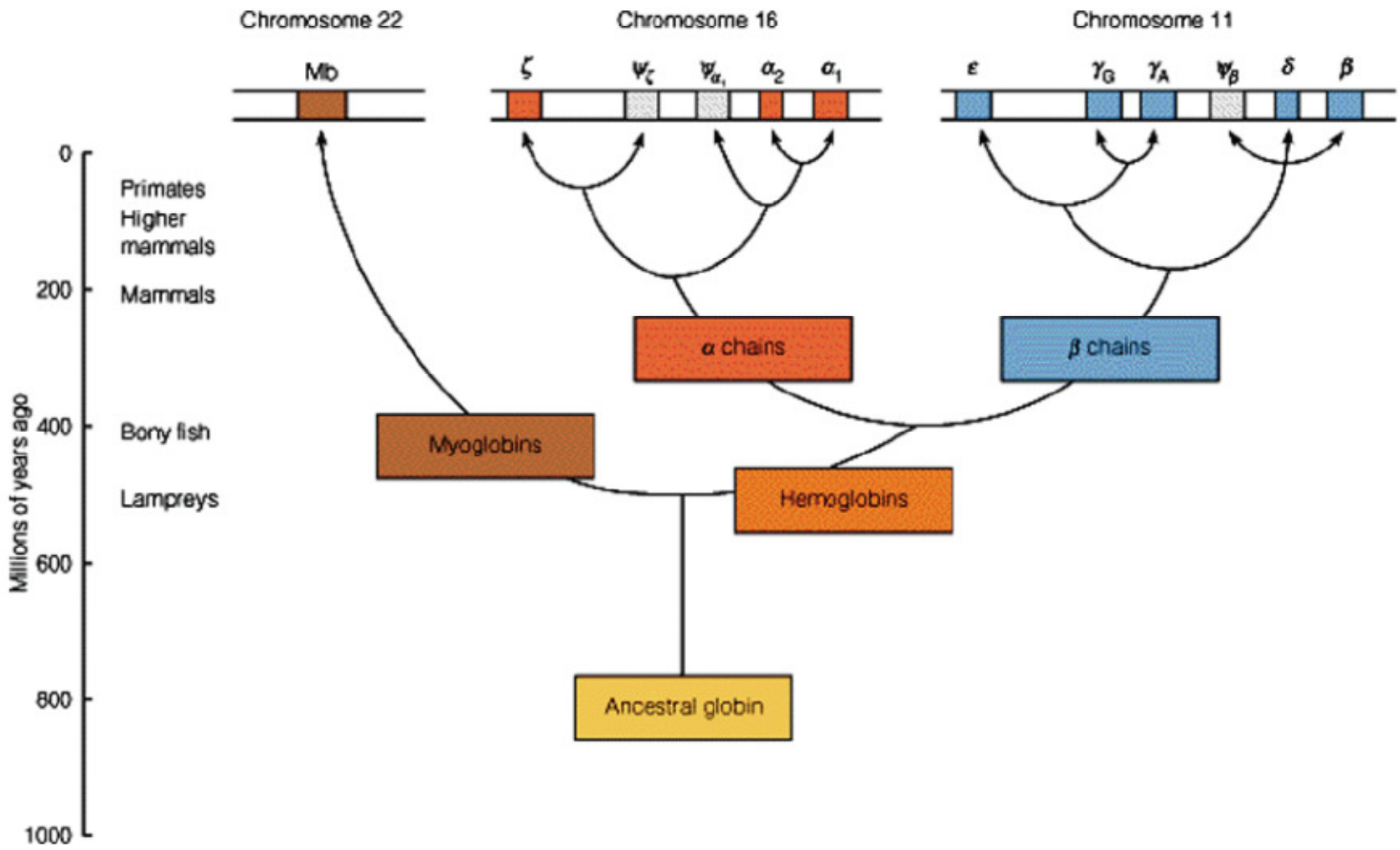
Osnovi filogenetskog stabla

- Listovi predstavljaju objekte (gene, vrste, individue / sojeve)
  - Termin takson (taxon / taxa) označava vrstu i šire klasifikacije organizama
- Unutrašnji čvorovi su hipotetički preci
- U stablu sa korenom, putanja od korena do čvora predstavlja evolucionu putanju
  - Koren predstavlja zajedničkog pretka
- Stablo bez korena predstavlja odnose među objektima, ali ne i usmerene evolucione putanje

Motivacija

- Zašto konstruisati stabla
  - Da bi se razumeli evolucioni odnosi vrsta
  - Da bi se razumelo kako su različite funkcije evoluirale
  - Da bi se identifikovalo šta je najočuvanije / najvažnije u nekoj klasi sekvenci

Primer stabla gena: globini (vrsta proteina)



Primer stabla vrsta: babuni

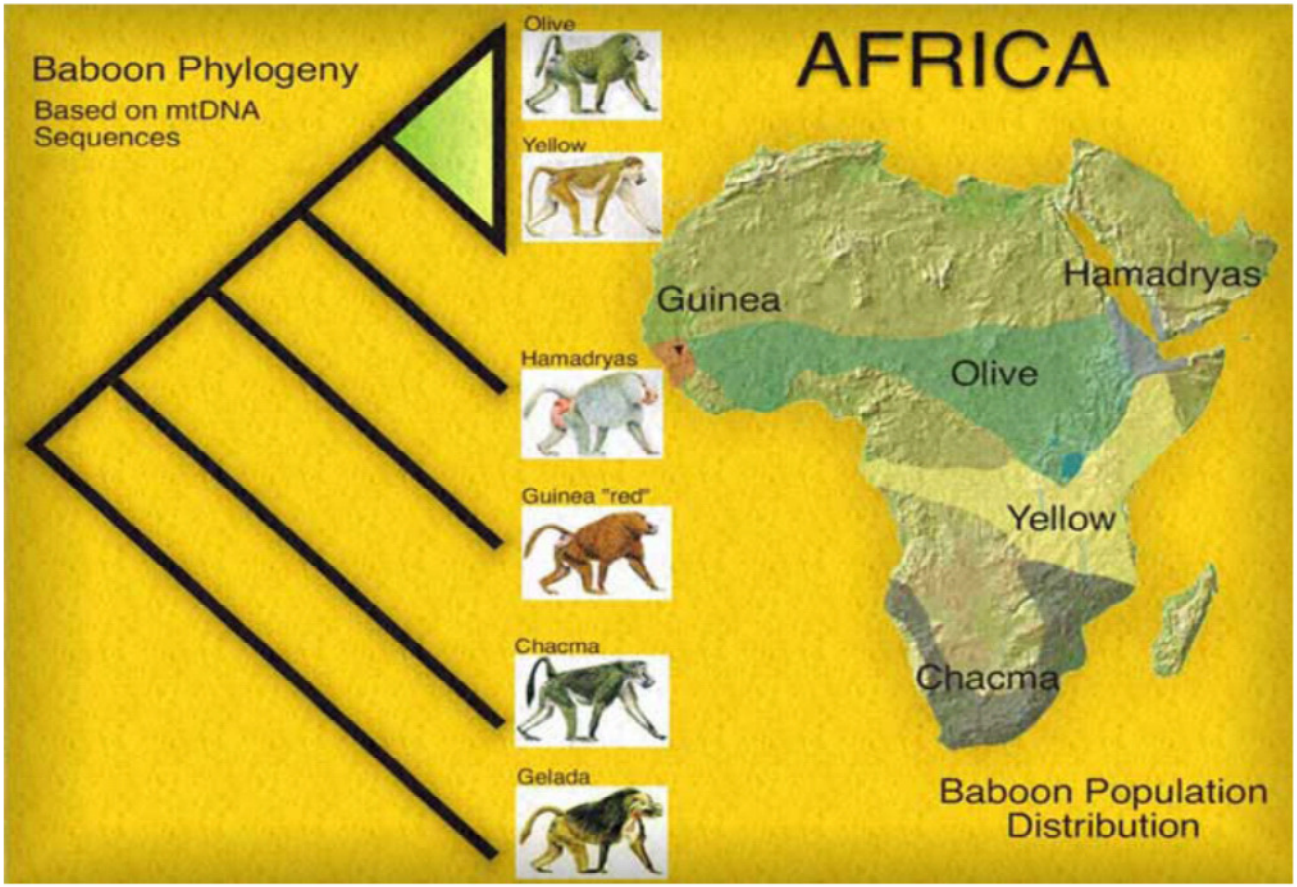


Image from Southwest National Primate Research Center

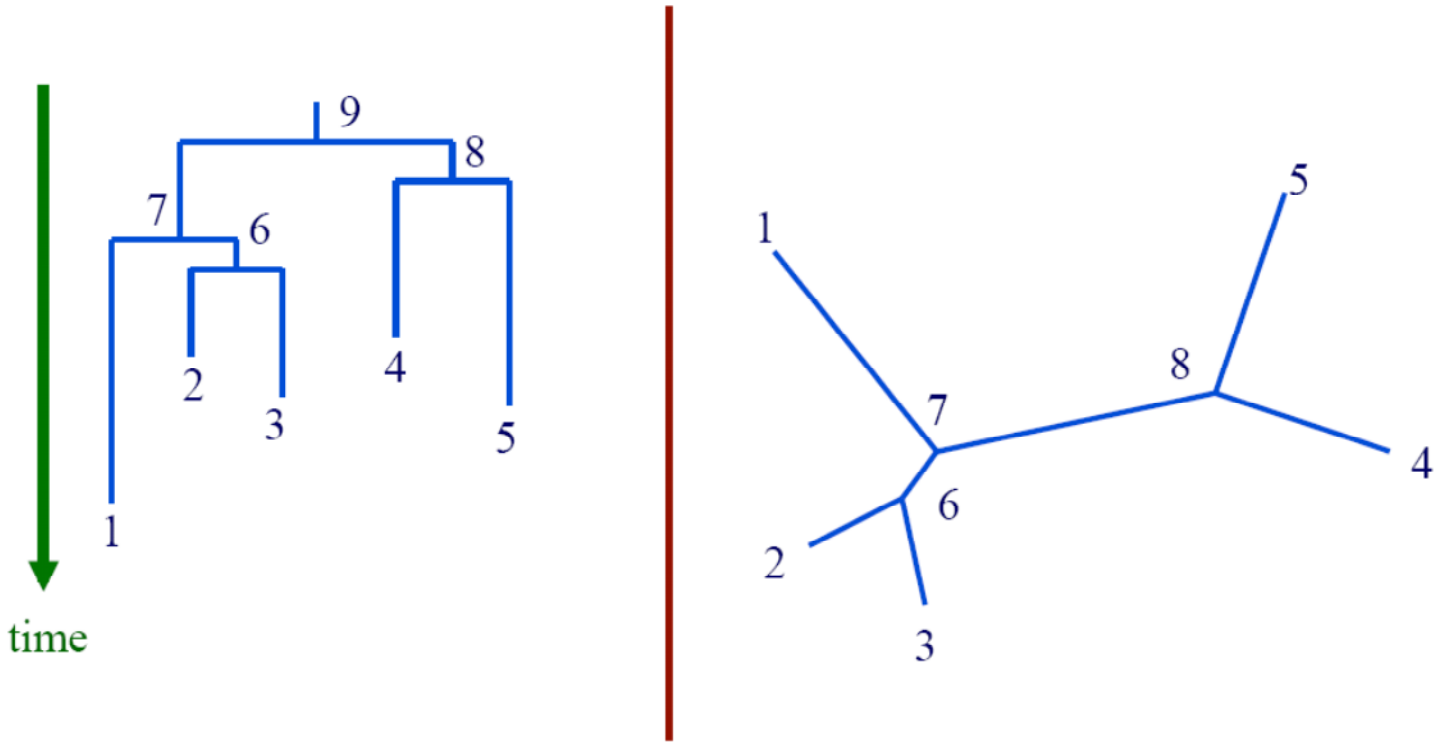
Broj mogućih stabala

- Razmatraju se samo binarna stabla, tj. u svakom čvoru grananja susreću se tri grane
- Pravo filogenetsko stablo ima koren, ili zajednički predak svih sekvenci
- U korenom stablu sa n listova ima 2n-1 čvorova (n listova + n-1 unutrašnjih čvorova) i 2n-2 grane (bez grane iz korena)
- Oznaka 2n-1 rezervisana je za koreni čvor
- Nekoreno stablo sa n listova ima 2n-2 čvora i 2n-3 grane; dodavanjem korena na bilo koju granu jednog nekorenog stabla dobije se koreno stablo; zato za jedno nekoreno stablo postoji 2n-3 korenih stabala
- Korenih stabala sa n listova ima 2n-3 puta više nego nekorenih
- Koliko ima nekorenih stabala sa n listova?
- Umesto korena – grana: ima 3*5*...*(2n-5)=(2n-5)!! nekorenih stabala sa n listova (sekvenci) (i (2n-3)!! korenih stabala)

Broj mogućih stabala

# taxa (n)	# unrooted trees	# rooted trees
4	3	15
5	15	105
6	105	945
8	10,395	135,135
10	2,027,025	34,459,425

Koreno vs. nekoreno stablo



Podaci za izgradnju stabla

- Stabla mogu da se konstruišu iz raznih tipova podataka
  - Zasnovani-na-rastojanju: mere udaljenosti između vrsta / gena
  - Zasnovani na svojstvima: morfološka svojstva (npr. broj nogu)
  - Redosled-gena: linearni redosled ortologih (specijacija) gena u datim genomima

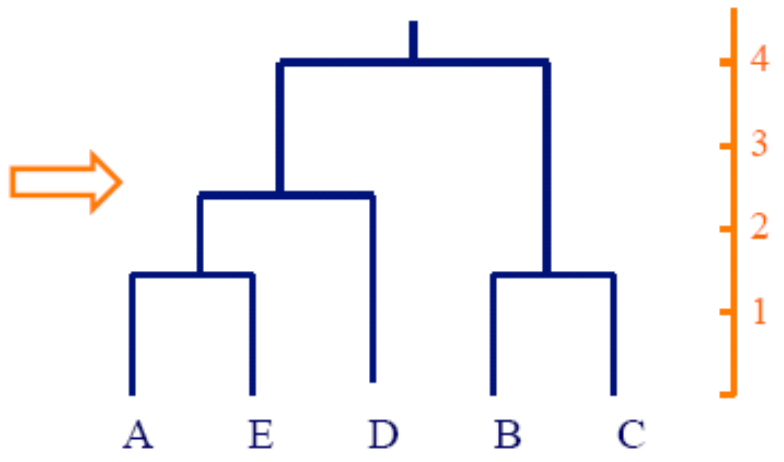
Pristupi filogenetskom stablu

- Tri opšta tipa metoda
  - Rastojanje: naći stablo koje opravdava ocenjena evoluciona udaljenja
  - Parsimonija: naći stablo koje zahteva najmanji broj promena da bi objasnilo podatke
  - Najveća verovatnoća: naći stablo koje maksimizuje verovatnoću podataka

Pristupi zasnovani na rastojanju

- Neka je data matrica M_{nxn} gde je M_{ij} rastojanje između taksona *i* i *j*
- Izgraditi težinsko stablo tako da rastojanja između listova *i* i *j* odgovaraju M_{ij}

	A	B	C	D	E
A	0	8	8	5	3
B		0	3	8	8
C			0	8	8
D				0	5
E					0



Rastojanja

- Obično se dobiju iz poravnanja sekvenci
- f_{ij} = #neslaganja/(#slaganja + #neslaganja)

u poravnanju sekvenci *i* i *j*

- dist(i,j) = f_{ij}
- Svojstva metrike rastojanja

$$\mathrm{dist}(x_i, x_j) \geq 0$$

$$\mathrm{dist}(x_i, x_i) = 0$$

$$\mathrm{dist}(x_i, x_j) = \mathrm{dist}(x_j, x_i)$$

$$\mathrm{dist}(x_i, x_j) \leq \mathrm{dist}(x_i, x_k) + \mathrm{dist}(x_k, x_j)$$

Ultrametrični podaci (“ultrametric data”)

- Pretpostavka o *molekulskom satu*: divergencija sekvenci se događa po istoj stopi u svim tačkama stabla

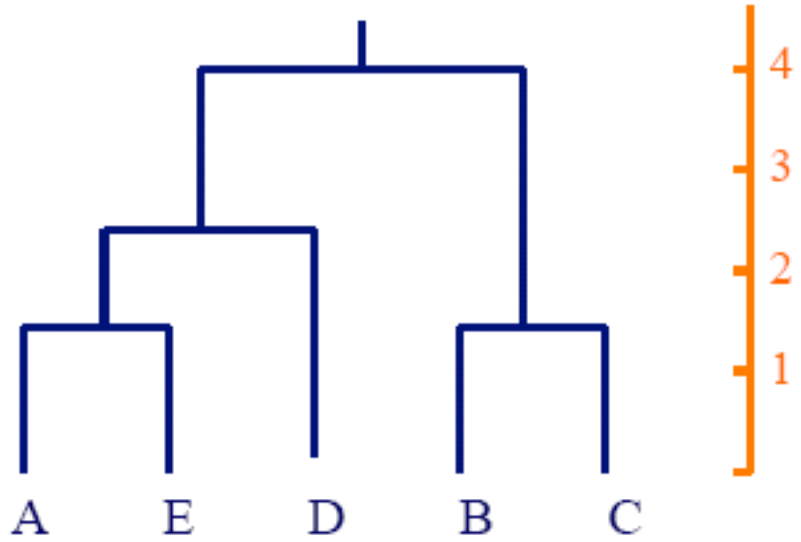
- Pretpostavka nije sasvim tačna (drugi faktori utiču)

- Ali ako je tačna, za podatke se kaže da su *ultrametrični*

Ultrametrični podaci

- Za ultrametrične podatke važi: za svaku trojku sekvenci *i,j,k*, rastojanja su ili sva jednaka ili su dva jednaka a treće manje

	A	B	C	D	E
A	0	8	8	5	3
B		0	3	8	8
C			0	8	8
D				0	5
E					0



UPGMA metod ("**U**nweighted **P**air **G**roup **M**ethod using **A**rithmetic Averages")

- Procedura klasterovanja (Sokal & Michener 1958)
- Ako su dati ultrametrični podaci, UPGMA će rekonstruisati stablo T koje je konzistentno sa podacima
- Osnovna ideja:
  - Iterativno odabirati dva taksona / klastera i spajati ih
  - Kreirati novi čvor u stablu za spojene klastere
  - Rastojanje d_{ij} između klastera C_i i C_j taksona definiše se kao

$$d_{ij} = \frac{1}{|C_i||C_j|} \sum_{p \in C_i, q \in C_j} d_{pq}$$

  - (prosečno rastojanje između para taksona iz svakog klastera)

UPGMA algoritam

- Dodeliti svaki takson sopstvenom klasteru
- Definisati jedan list za svaki takson; postaviti ga na visinu 0
- Sve dok ima više od dva klastera
  - Odrediti dva klastera i,j sa najmanjim d_{ij}
  - Definisati novi klaster $C_k = C_i \cup C_j$
  - Definisati čvor *k* sa potomcima *i* i *j*; postaviti ga na visinu $d_{ij}/2$
  - Zameniti klastere *i* i *j* klasterom *k*
  - Izračunati rastojanje između *k* i drugih klastera
- Spojiti poslednja dva klastera, *i* i *j*, korenom na visini $d_{ij}/2$

UPGMA

- Ako je dat novi klaster C_k, dobijen spajanjem C_i i C_j ($C_k = C_i \cup C_j$),
  možemo da izračunamo rastojanje između C_k i bilo kog drugog klastera C_l prema sledećoj formuli (proveriti za vežbu – koristiti definiciju d_{ij}):

$$d_{kl} = \frac{d_{il} \mid C_i \mid + d_{jl} \mid C_j \mid}{\mid C_i \mid + \mid C_j \mid}$$

UPGMA primer

initial state

	A	B	C	D	E
A	0	8	8	5	3
B		0	3	8	8
C			0	8	8
D				0	5
E					0

after one merge

	AE	B	C	D
AE	0	8	8	5
B		0	3	8
C			0	8
D				0

UPGMA primer (nast.)

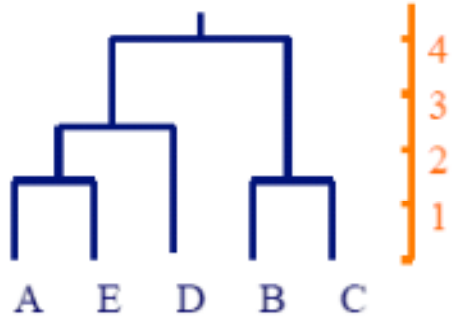
after two merges

	AE	BC	D
AE	0	8	5
BC		0	8
D			0

after three merges

	AED	BC
AED	0	8
BC		0

final state

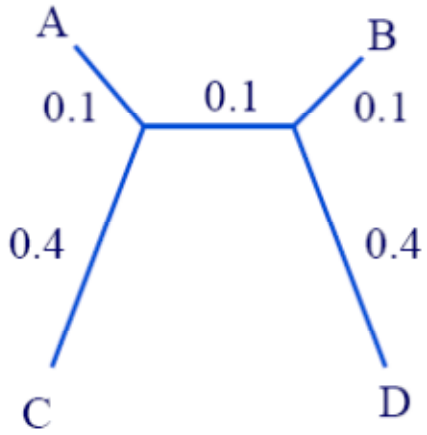


Spajanje suseda (Neighbor Joining)

- Za razliku od UPGMA
  - Ne polazi od pretpostavke molekulskog sata
  - Proizvodi nekorena stabla
- Podrazumeva *aditivnost*: rastojanje između para listova je zbir dužina grana koje ih povezuju
- Kao i UPGMA, konstruiše stablo iterativnim spajanjem podstabala
- Dve ključne razlike
  - Kako se bira par podstabala koja će se spajati u svakoj iteraciji
  - Kako se ažuriraju rastojanja posle svakog spajanja

Odabir parova čvorova za spajanje u NJ

- Na svakom koraku, odabiramo par čvorova za spajanje; da li da odaberemo par sa minimalnim d_{ij}?
- Ako stablo izgleda kao na slici i mi odaberemo prvi par čvorova za spajanje:



$$d_{AB} = 0.3$$

$$d_{AC} = 0.5$$

- Pogrešna odluka da se spoje A i B, zbog dugačke grane lista C: treba razmotriti rastojanje parova drugih listova

Odabir parova čvorova za spajanje u NJ

- Da bi se "kompenzovale" dugačke grane, bira se par za spajanje na bazi D_{ij}

[Saitou & Nei '87; Studier & Keppler '88]

$$D_{ij} = d_{ij} - (r_i + r_j)$$

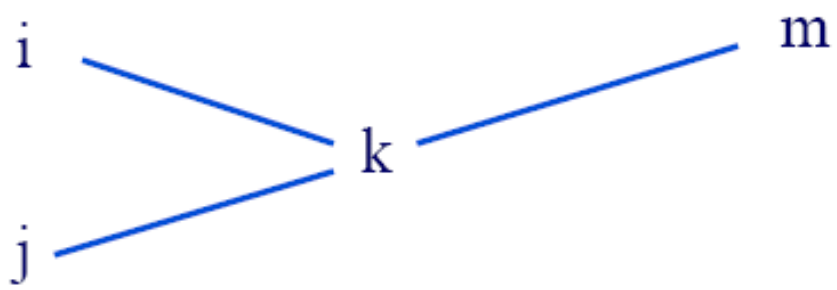
$$r_i = \frac{1}{|L| - 2} \sum_{k \in L} d_{ik}$$

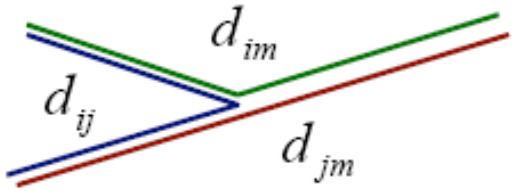
- L je skup listova; r_i je srednje udaljenje lista r_i od svih listova stabla različitih od *i, j*; slično r_j

Ažuriranje rastojanja u NJ

- Kada je dat novi unutrašnji čvor *k*, roditelj listova - suseda *i* i *j*, rastojanje čvora *k* od nekog drugog lista *m* dato je sa (zbog aditivnosti; v. sliku dole):

$$d_{km} = \frac{1}{2}(d_{im} + d_{jm} - d_{ij})$$

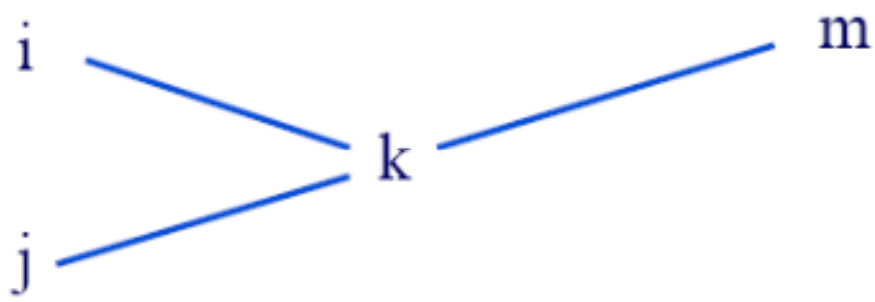




Ažuriranje rastojanja u NJ

- Može da se sračuna rastojanje lista *i* od roditeljskog čvora *k* na isti način (preko *nekog* lista m):

$$d_{ik} = \frac{1}{2}(d_{ij} + d_{im} - d_{jm})$$



$$d_{jk} = d_{ij} - d_{ik}$$

Ažuriranje rastojanja u NJ

- Uopštenje: razmatrajmo rastojanja lista *i* od roditeljskog čvora *k* preko *svih* drugih listova

$$d_{ik} = \frac{1}{2}(d_{ij} + r_i - r_j)$$

- gde je $r_i = \frac{1}{|L|-2} \sum_{m \in L} d_{im}$

- a L je skup listova

Neighbor Joining algoritam

- Definisati stablo T kao skup listova
- L=T
- Sve dok ima više od dva podstabla u T
  - Odabrati par i,j u L sa najmanjim D_{ij}
  - Dodati u T novi čvor k koji spaja *i* i *j*
  - Odrediti nova rastojanja

$$d_{ik} = \frac{1}{2}\left(d_{ij} + r_i - r_j\right)$$

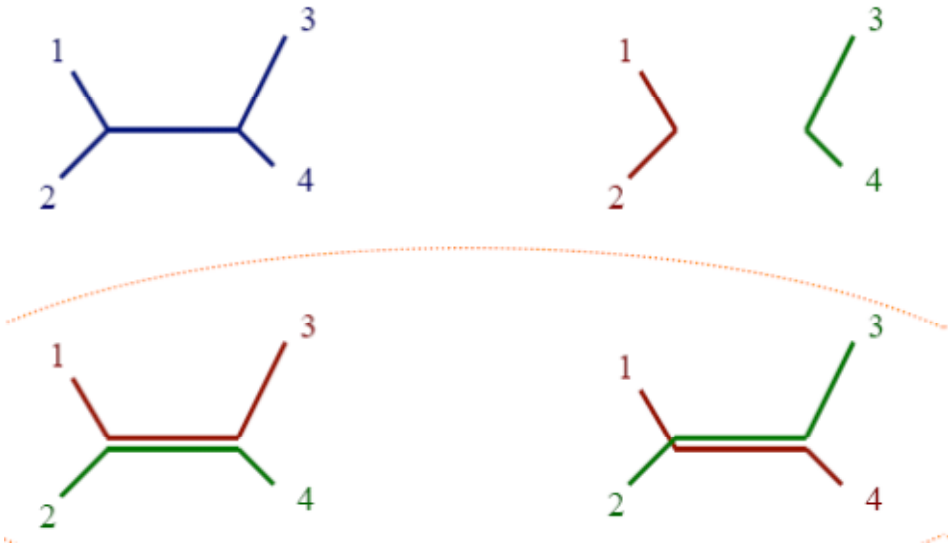
$$d_{jk} = d_{ij} - d_{ik}$$

$$d_{km} = \frac{1}{2}\left(d_{im} + d_{jm} - d_{ij}\right) \text{ for all other } m \text{ in } L$$

  - Ukloniti *i* i *j* iz L i uneti *k* (tretirati ga kao list)
- Spojiti dva preostala podstabla, *i* i *j* granom dužine d_{ij}

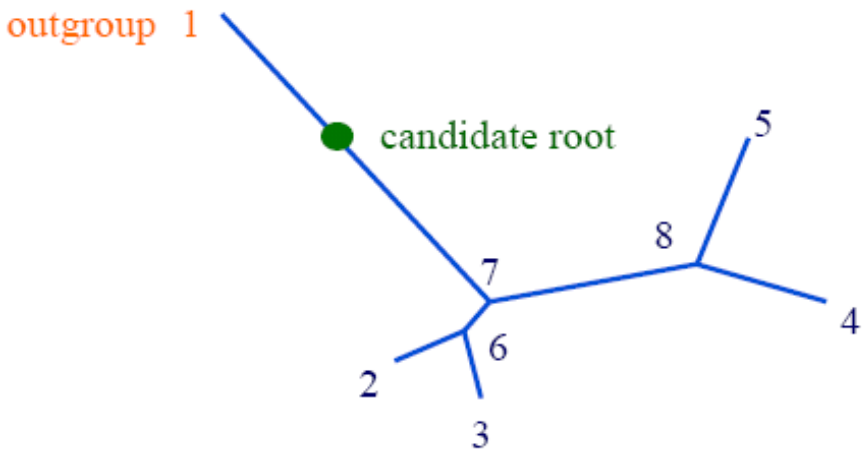
Provera aditivnosti

- Za svaki skup od četiri lista, i,j,k i l, dva od rastojanja $d_{ij}+d_{kl}$, $d_{ik}+d_{jl}$ i $d_{il}+d_{jk}$ moraju da budu jednaka i ne manja od trećeg



Dodavanje korena stablu

- Nalaženje korena u nekorenom stablu nekad se izvodi korišćenjem *outgroup ("stranca")*
- Outgroup: vrsta za koju se zna da je udaljenija od preostalih vrsta nego što su one među sobom
- Grana koja spaja outgroup sa ostatkom stabla je najbolji kandidat za poziciju korena



Zaključak o metodama zasnovanim na rastojanju

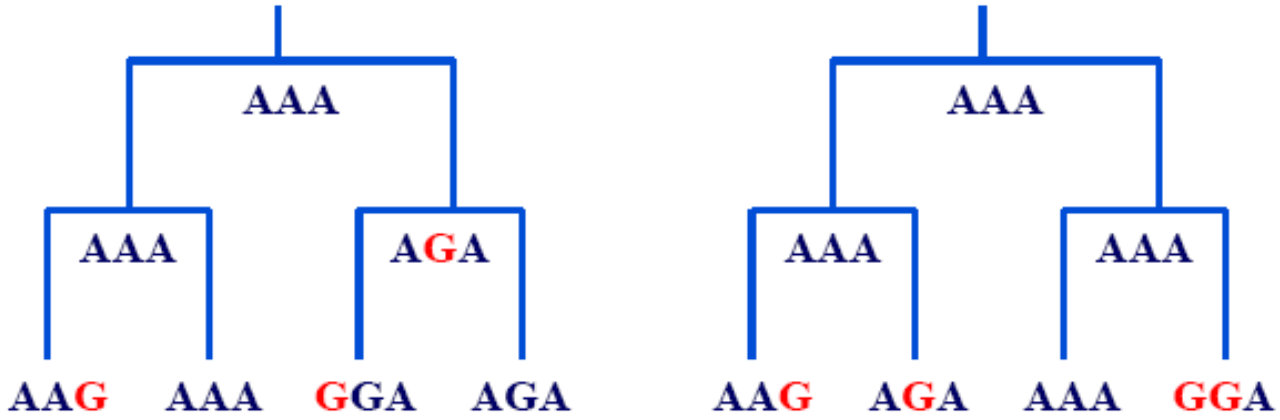
- Ako su dati podaci o rastojanjima ultrametrični, UPGMA će identifikovati korektno, koreno stablo
- Ako su podaci aditivni, NJ će identifikovati korektno, nekoreno stablo
- Inače, metode mogu da ne rekonstruišu korektno stablo, ali mogu još uvek da se koriste kao razumne heuristike

Pristupi zasnovani na parsimoniji

- Ulaz: podaci zasnovani na svojstvima
- Pronaći stablo koje objašnjava podatke minimalnim brojem promena (supstitucija)
- Fokus je na nalaženju korektne topologije stabla, ne na proceni dužine grana

Primer parsimonije

- Razna stabla mogu da objasne filogeniju sekvenci AAG, AAA, GGA, AGA, na primer:



- Parsimonija preferira prvo stablo zato što ono zahteva manje supstitucija

Pristupi bazirani na parsimoniji

- Obično uključuju dve zasebne komponente
  - 1. proceduru za nalaženje minimalnog broja promena potrebnih da se objasne podaci (za datu topologiju stabla)
  - 2. pretragu kroz prostor stabala

Nalaženje minimalnog broja promena za *dato* stablo

- Osnovne pretpostavke
  - Svako stanje (npr. nukleotid, amino kiselina) može da pređe u bilo koje drugo stanje
  - “cene” ovih promena su uniformne
  - Pretpostavka je da su *pozicije sekvence nezavisne*
    - Može da se računa minimalni broj promena za svaku poziciju sekvence nezavisno

Nalaženje minimalnog broja promena za dato stablo

- Metod grube sile
  - Za svaku moguću dodelu stanja unutrašnjim čvorovima
    - Izračunati broj promena za tu dodelu
  - Saopštiti nađeni minimalni broj promena
  - Vreme izvršavanja: $O(Nk^N)$
    - K = broj mogućih stanja svojstva (npr. 4 za DNK)
    - N = broj listova

Fičov algoritam

- Fitch, 1971:

- 1. obići stablo od listova do korena i odrediti skup mogućih stanja (npr. nukleotida) za svaki unutrašnji čvor

- 2. obići stablo od korena do listova i izabrati jedinstvena stanja za unutrašnje čvorove

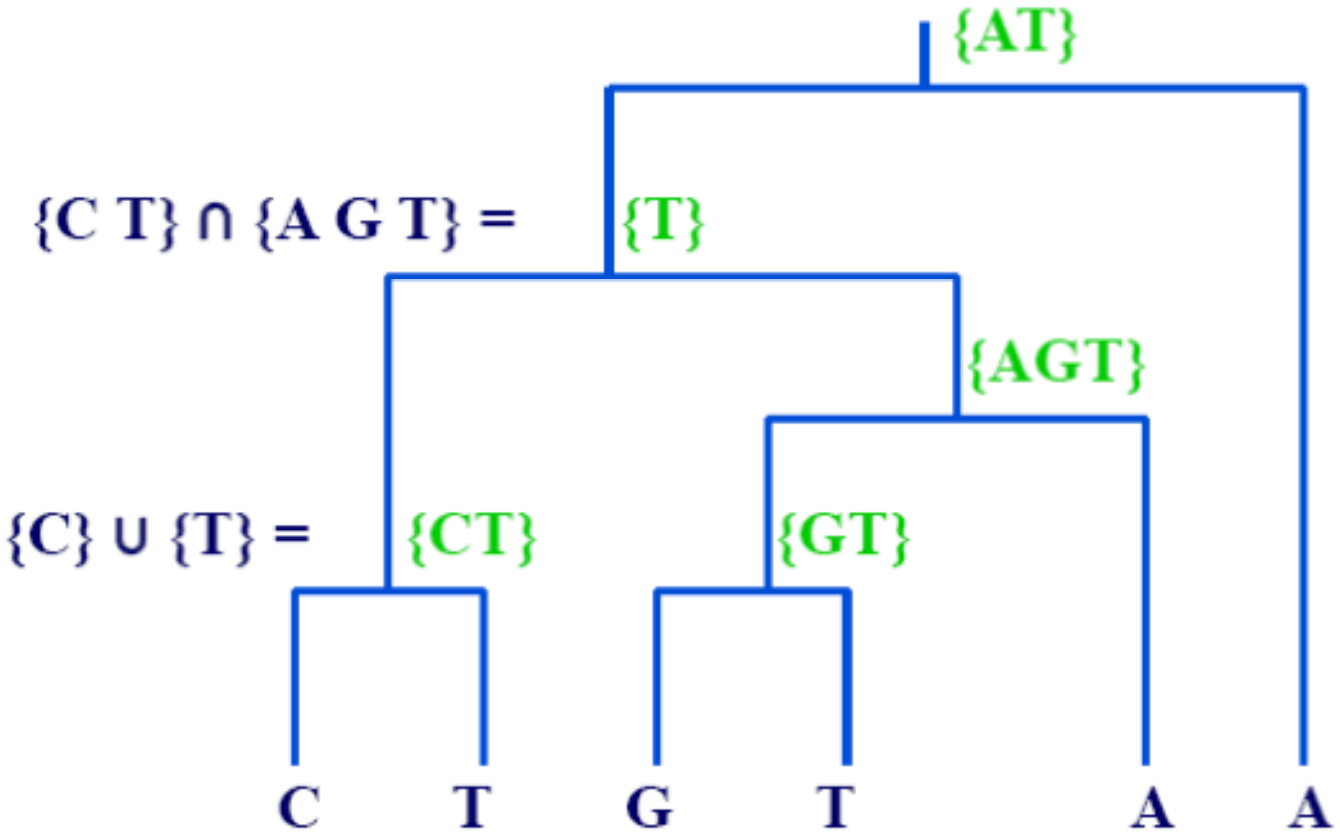
Fičov algoritam: korak 1
Moguća stanja za unutrašnje čvorove

- Obići stablo u postfiksnom poretku (post-order) – od listova ka korenu
- Odrediti moguća stanja R_i unutrašnjeg čvora *i* sa potomcima *j* i *k*:

$$R_i = \begin{cases} R_j \cup R_k, & \text{if } R_j \cap R_k = \varnothing \\ R_j \cap R_k, & \text{otherwise} \end{cases}$$

- Ovaj korak izračunava broj potrebnih promena
- # promena = # operacija unije

Fičov algoritam: korak 1
Primer

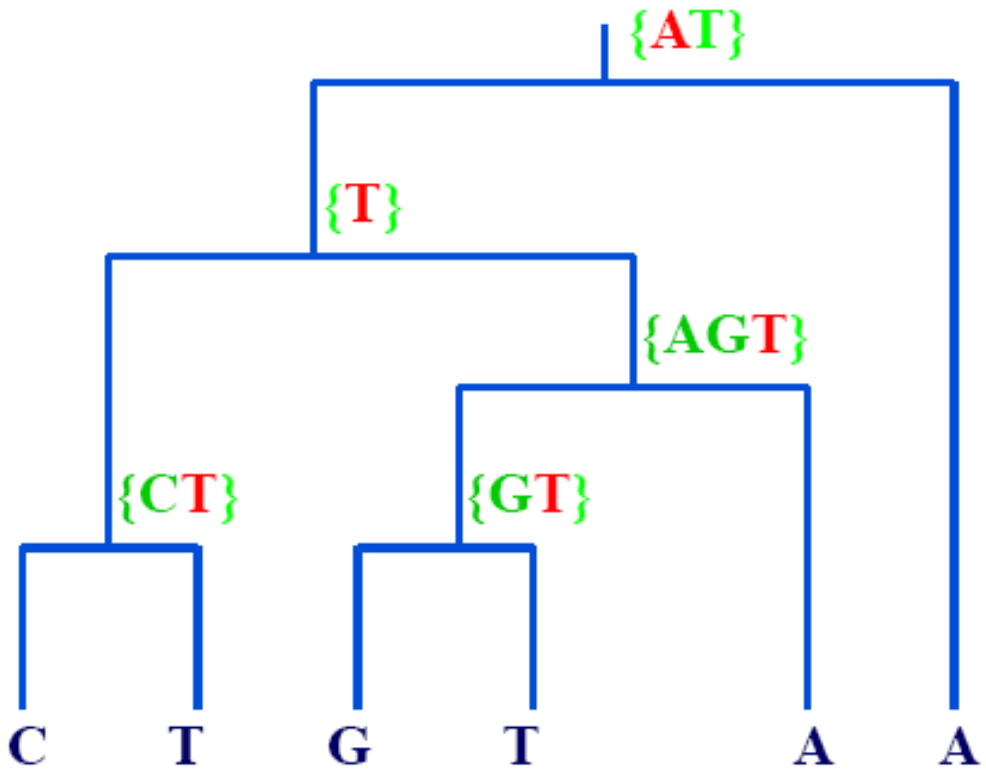


Fičov algoritam: korak 2
Izabrati stanja za unutrašnje čvorove

- Obići stablo u prefiksnom poretku (pre-order) – od korena do listova
- Izabrati stanje r_j unutrašnjeg čvora *j* sa prethodnikom *i*

$$r_j = \begin{Bmatrix} r_i, \text{ if } r_i \in R_j \\ \text{arbitrary state} \in R_j, \text{ otherwise} \end{Bmatrix}$$

Fičov algoritam: korak 2
Primer



Posebna tehnika: težinska parsimonija

- Stankoff & Cedergren, 1983

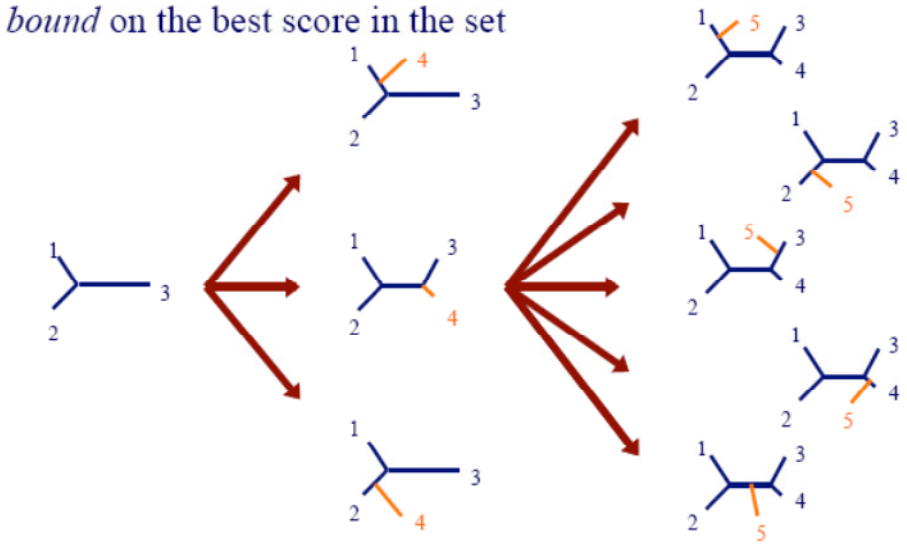
- Umesto pretpostavke da su sve promene stanja podjednako verovatne, koristiti različite cene S(a,b) za različite promene
- Svodi se na tradicionalnu (ne-težinsku) parsimoniju za S(a,a)=0 za svako a, i S(a,b)=1 za $a \neq b$.

Istraživanje prostora stabala

- Razmatrali smo kako da nađemo minimalni broj promena za datu topologiju stabla
- Potrebna je i procedura pretraživanja prostora topologija stabala
  - Kako se krećemo od jednog stabla do drugog?
  - Kako obezbeđujemo puno istraživanje prostora stabala?

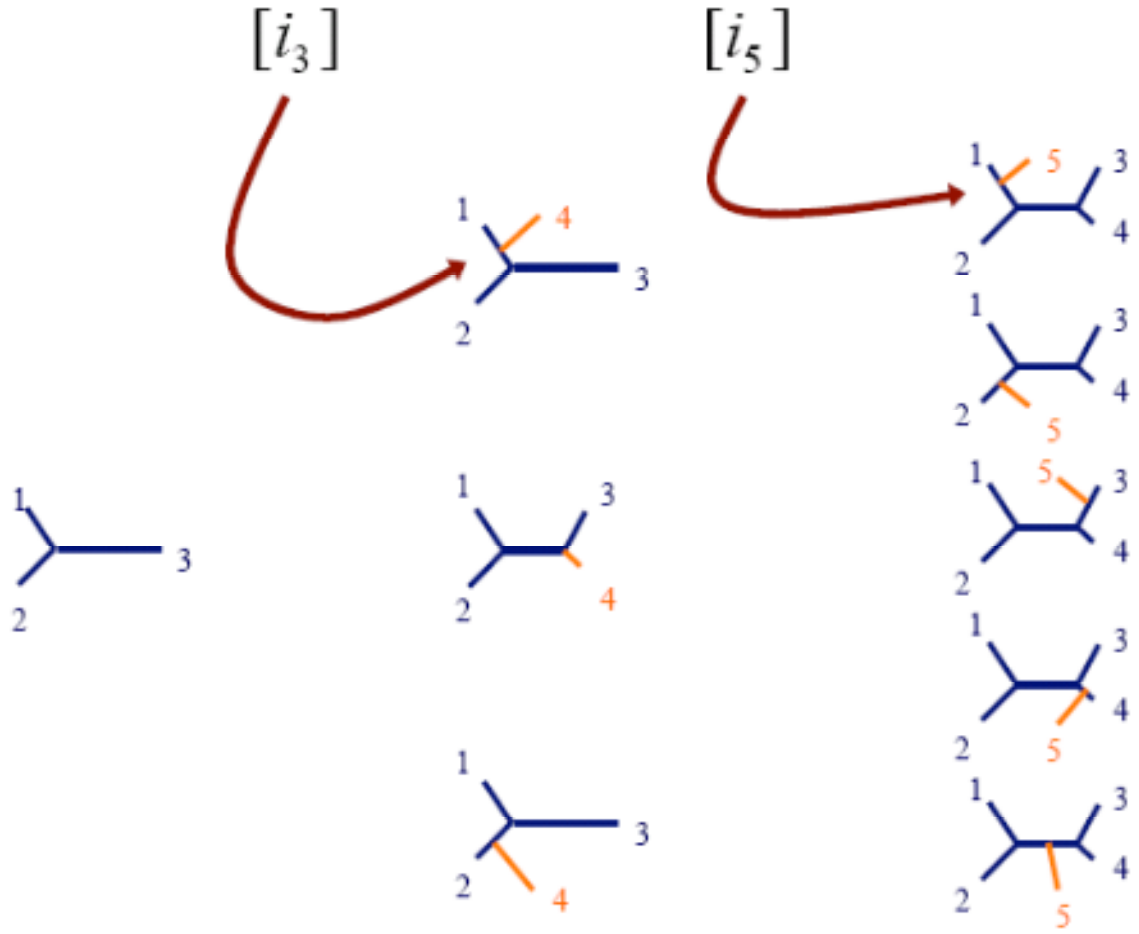
Tačna metoda: grananje i odsecanje

- Dovoljno je razmatrati nekorena stabla (bar jedna grana pod korenom nema supstituciju u optimalnom stablu)
- Svako parcijalno stablo predstavlja skup kompletnih stabala
- Skor parsimonije parcijalnog stabla obezbeđuje donju granicu najboljeg skora u skupu



- Obustavlja se dalja nadgradnja parcijalnog stabla čiji je skor veći od trenutno najmanjeg skora kompletno izgrađenog stabla

Implementiranje grananja i odsecanja - ilustracija



Implementiranje grananja i odsecanja

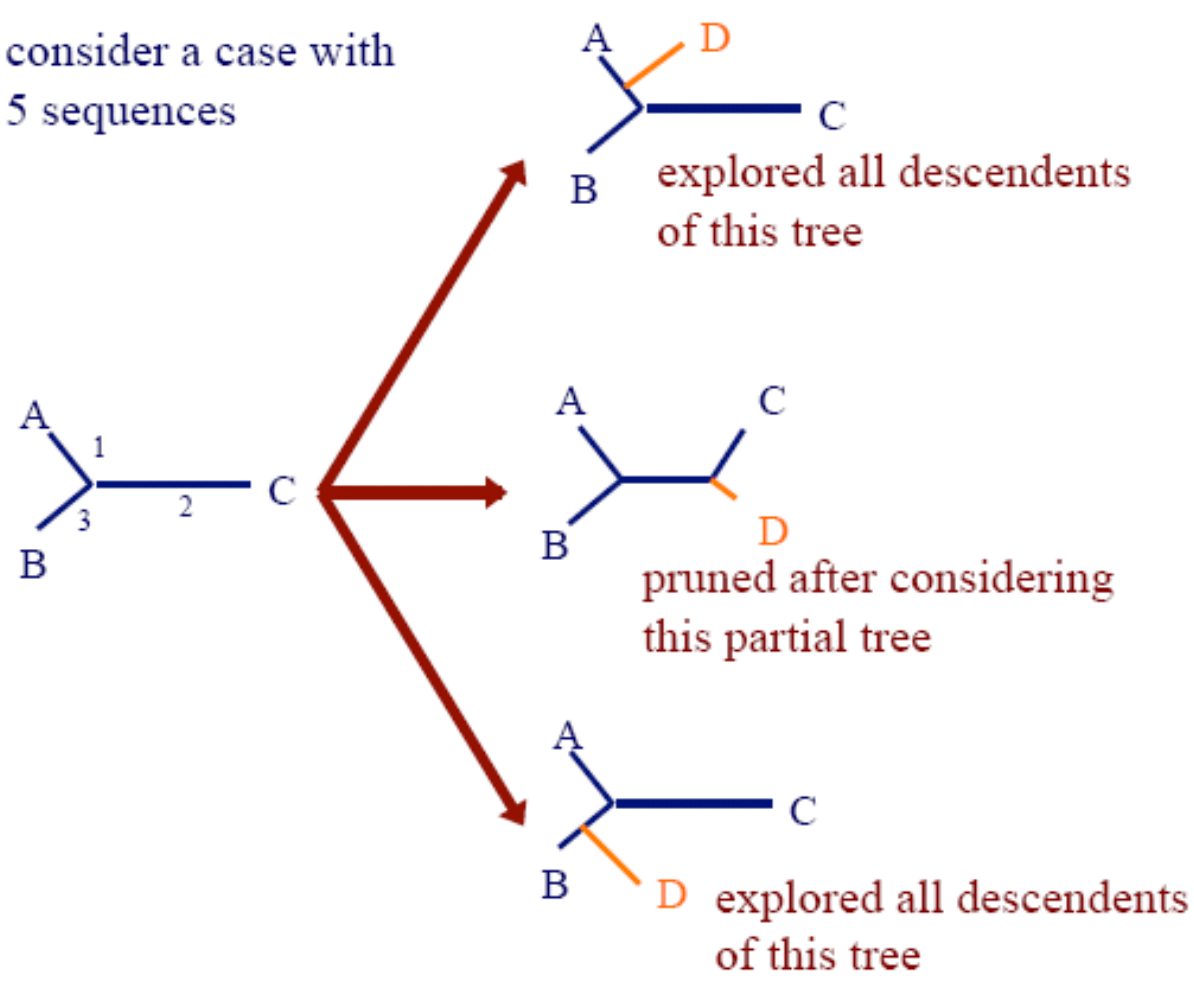
- Za n sekvenci, održavati niz brojača
  $[i_3][i_5][i_7]\ldots[i_{2n-5}]$,
  gde i_k uzima vrednosti 0..k
- Kompletno stablo je predstavljeno dodelom ne/nula vrednosti svim i_k
- i_k pokazuje, za parcijalno stablo sa *k* grana, na koju granu dodati grananje za sledeću sekvencu
- i_k=0 indikuje parcijalno stablo

Implementiranje grananja i odsecanja

- Vrteti brojače kroz njihove dopuštene brojeve kao odometre
  - Krajnji desni brojač se kreće najbrže
  - Kadgod je jedan brojač 0, svi brojači desno od njega moraju da budu 0
  - Proveriti cenu (parcijalnog) stabla na svakom otkucaju odometra
  - Odometar treba da preskoči kadgod se dogodi odsecanje

$$[i_3][i_5][i_7]\ldots[i_{2n-5}]$$

Implementiranje grananja i odsecanja



counters went:

$[i_3]$	$[i_5]$
1	0
1	1
1	2
1	3
1	4
1	5
2	0
3	0
3	1
3	2
3	3
3	4
3	5

Komentar o grananju i odsecanju

- To je kompletan metod pretraživanja
  - Garantuje se nalaženje optimalnog rešenja
  - Može da bude mnogo efikasnije od iscrpne pretrage
  - U najgorem slučaju, nije ni bolji

Komentar o izvođenju stabla

- Prostor pretraživanja može da bude veliki, ali
  - Optimalno stablo u nekim slučajevima može da se nađe efikasno
  - Heurističke metode mogu da se primene
  - Teško je oceniti izvedeno stablo: puna istina obično nije poznata
  - Neke novije metode koriste podatke zasnovane na linearnom uređenju ortologih gena duž hromozoma
  - Filogeneza za bakterije i viruse nije tako pravolinijska zbog bočnog transfera genetskog materijala