

بسم الله الرحمن الرحيم

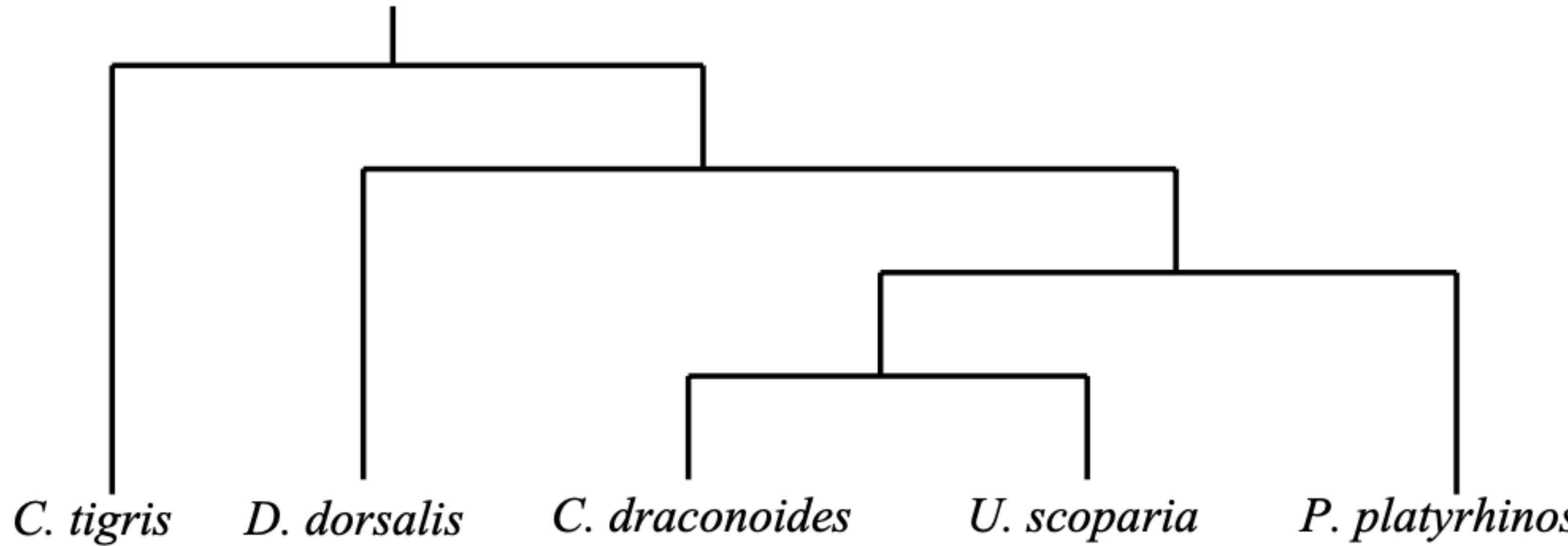
ڙنو ميڪ محاسباتي

جلسه ٤: بازسازی درخت تبارزایی (۱)

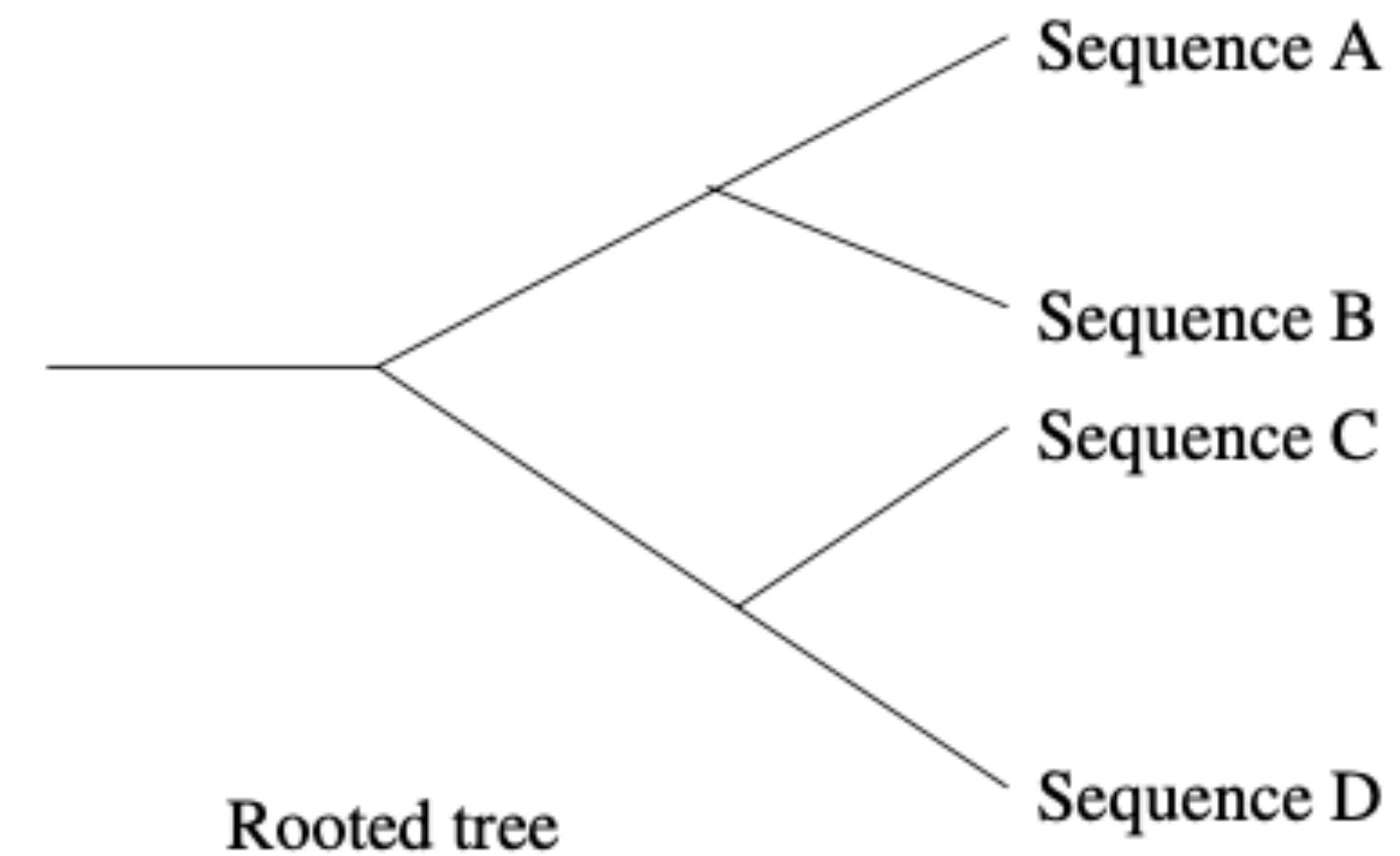
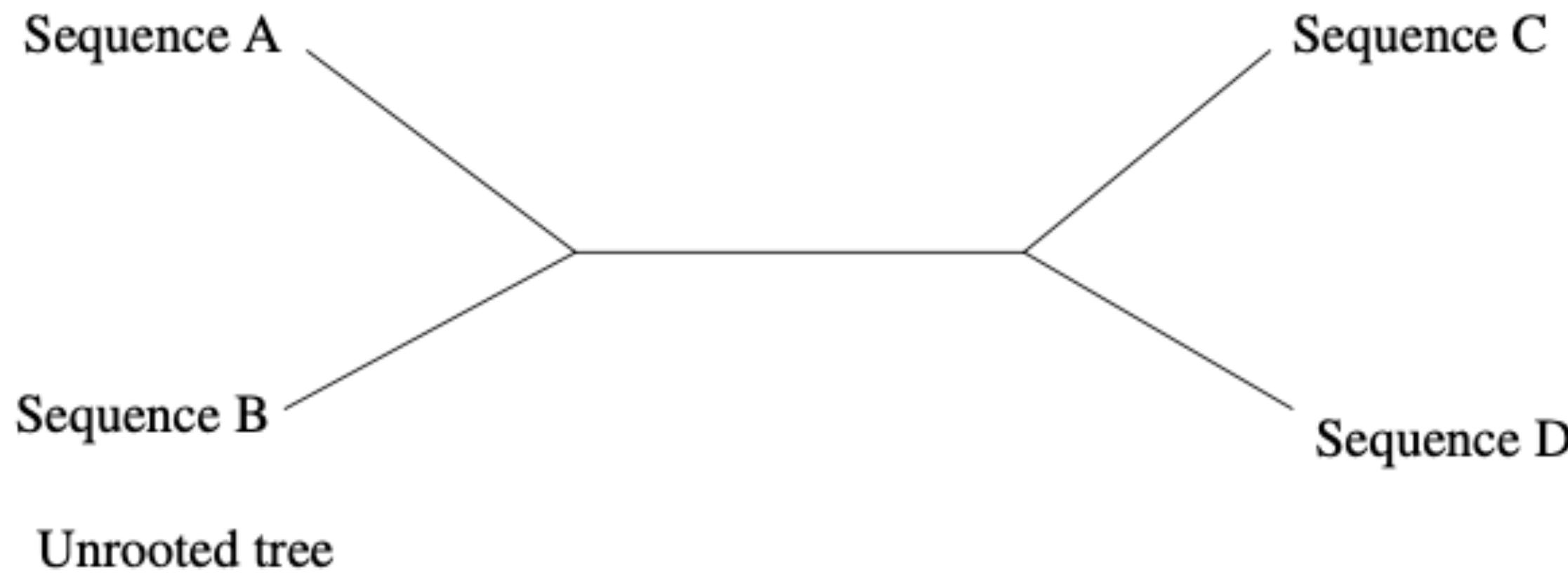
ترم پايز ١٤٠١-١٤٠٠



درخت تبارزایی، چیستی و چرا بی؟



درخت ریشه دار و درخت بی ریشه



تبديل درخت بـیـریشه به ریشه دار

- خارج از گروه
- مشکل خارج از گروه (دوری و نابجایی)
- وسط بزرگترین یال

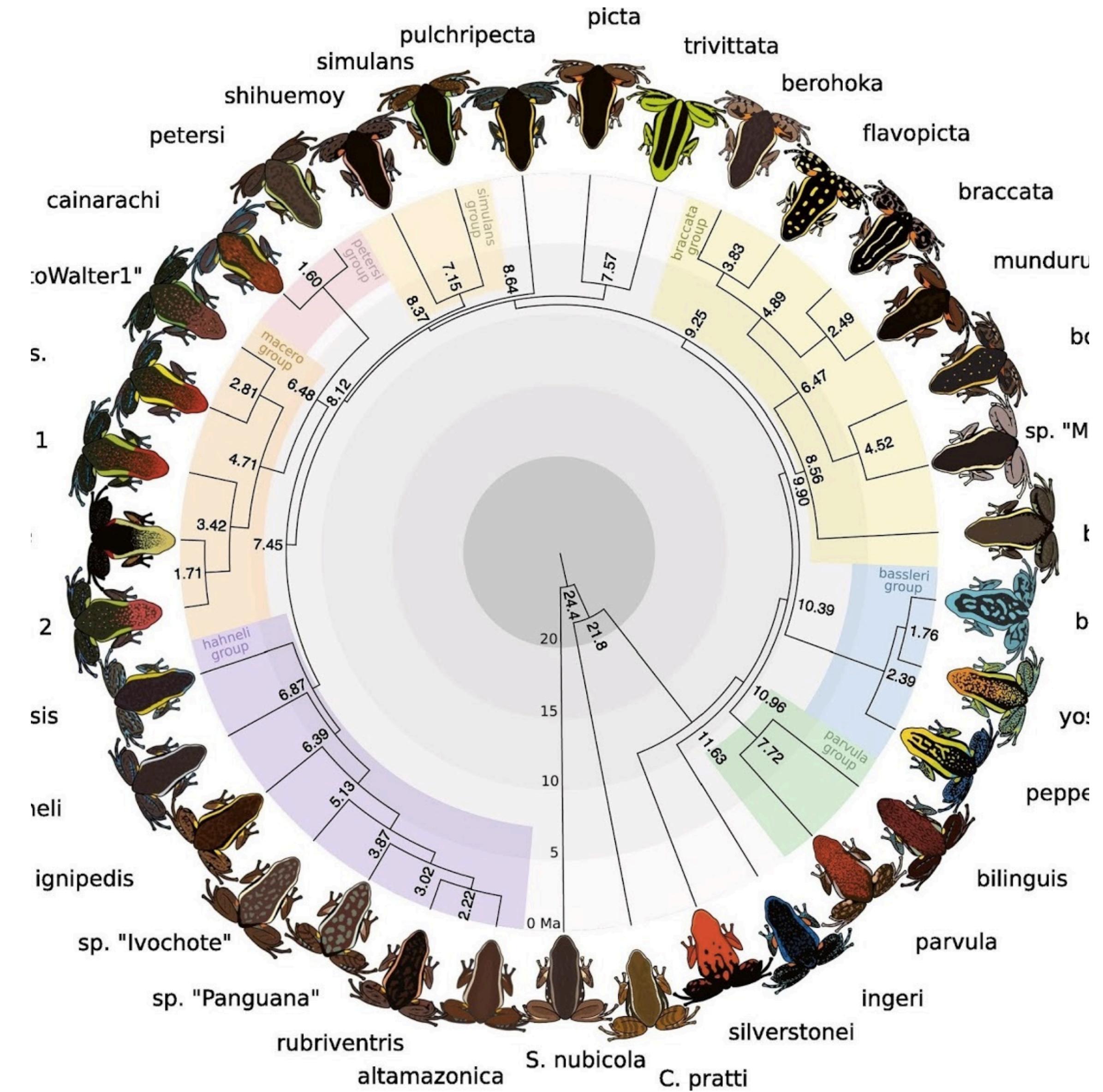
بازسازی درخت تبارزایی

- ورودی: ماتریس فاصله
- ورودی: ویژگی‌ها

معیار بهترین درخت

- بیشینه صرفه‌جویی
- یک مدل احتمالاتی
- معیارهای دیگر؟

بازسازی ویژگی مبنای درخت تبارزایی



بازسازی ویژگی مبنای درخت تبارزایی

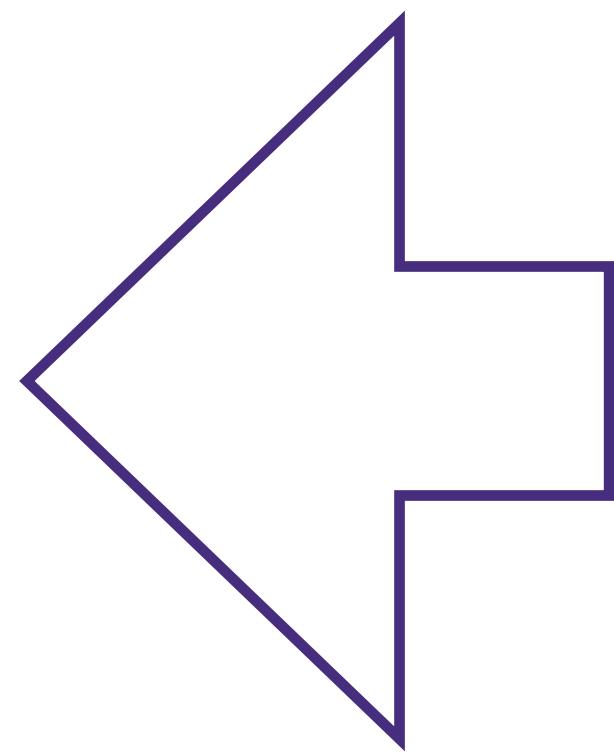
- ورودی:
- ویژگی ها
- ماتریس ویژگی ها
- مثلاً ماتریس توالی های ژنومی

مسئله صرفه جویانه ترین درخت

مسئله صرفه جویانه ترین درخت

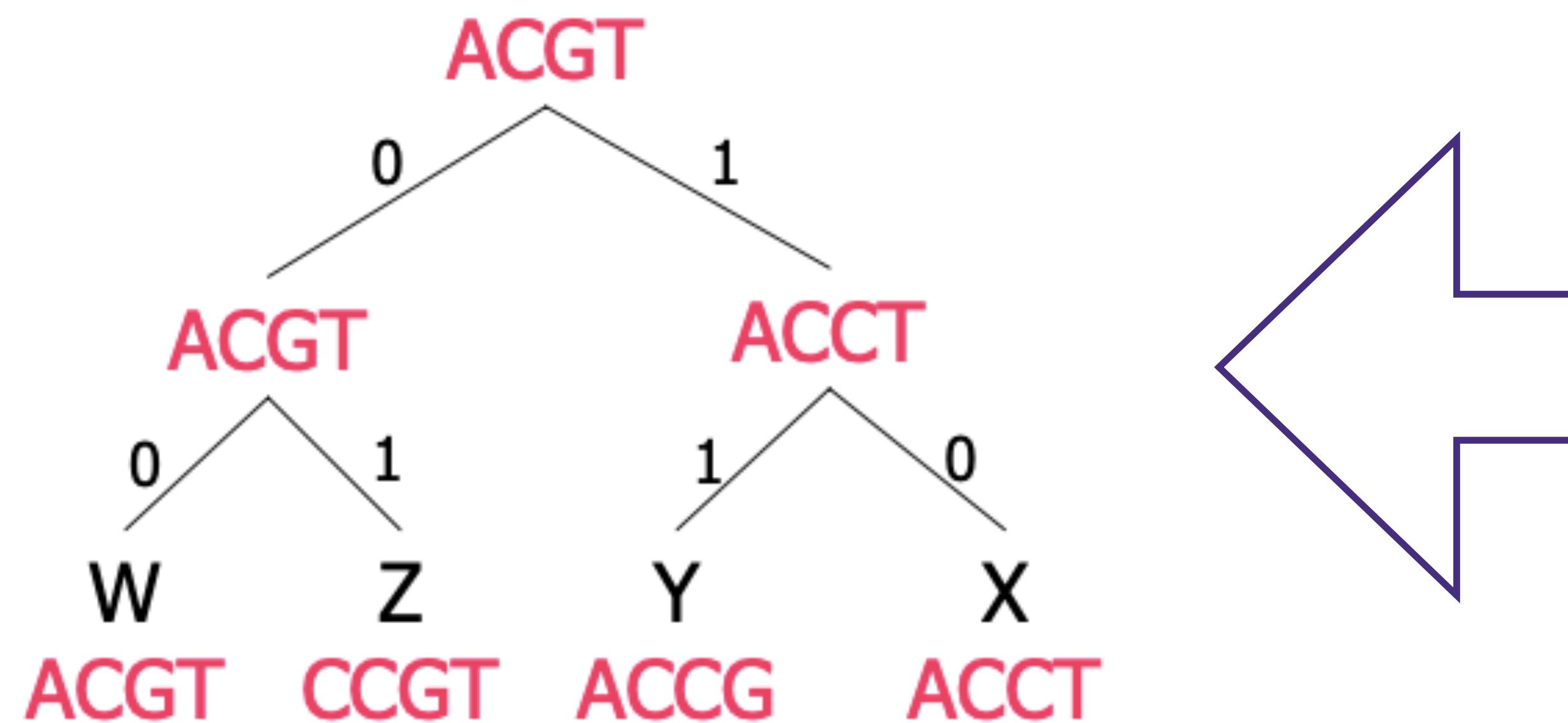
	1	2	3	4
W	A	C	G	T
X	A	C	C	T
Y	A	C	C	G
Z	C	C	G	T

مسئله صرفه جویانه ترین درخت



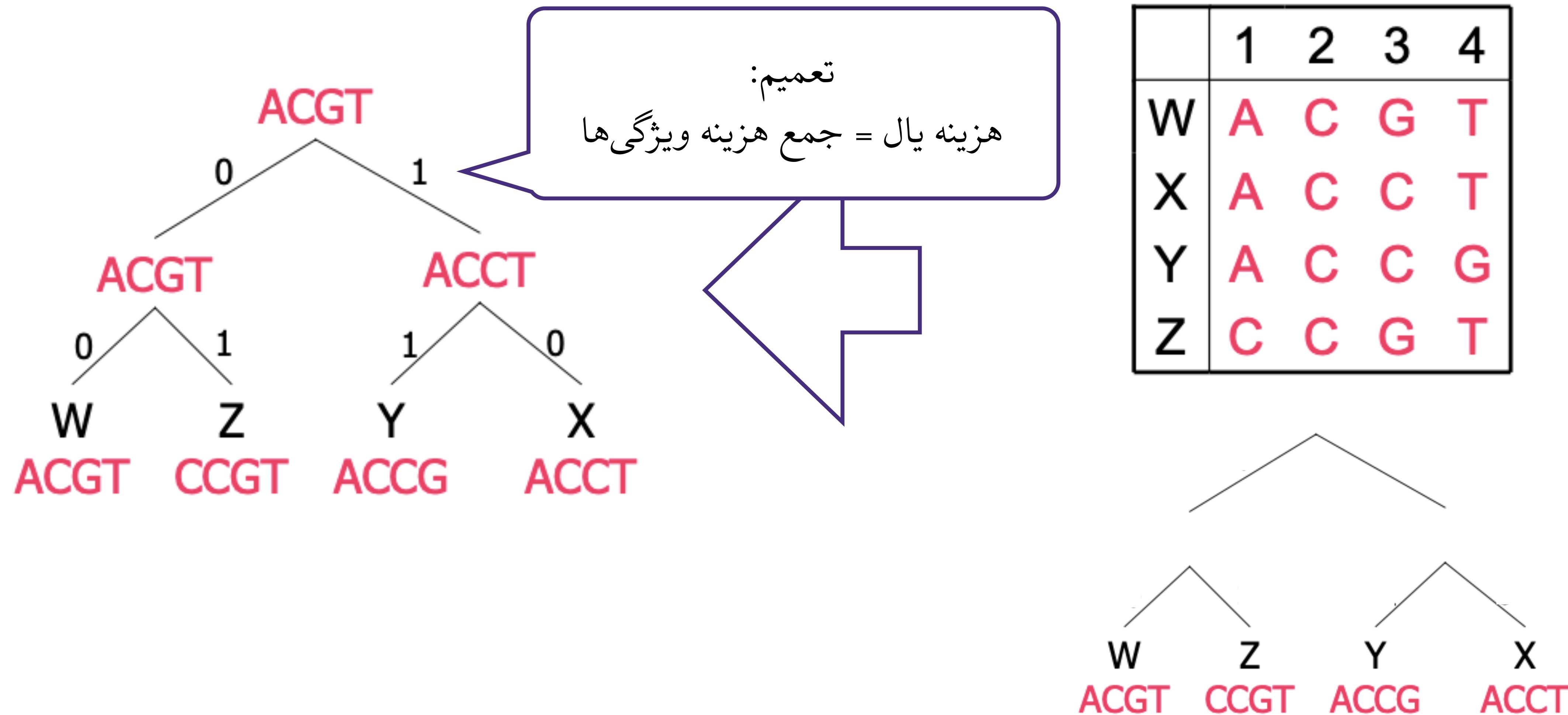
	1	2	3	4
W	A	C	G	T
X	A	C	C	T
Y	A	C	C	G
Z	C	C	G	T

مسئله صرفه جویانه ترین درخت

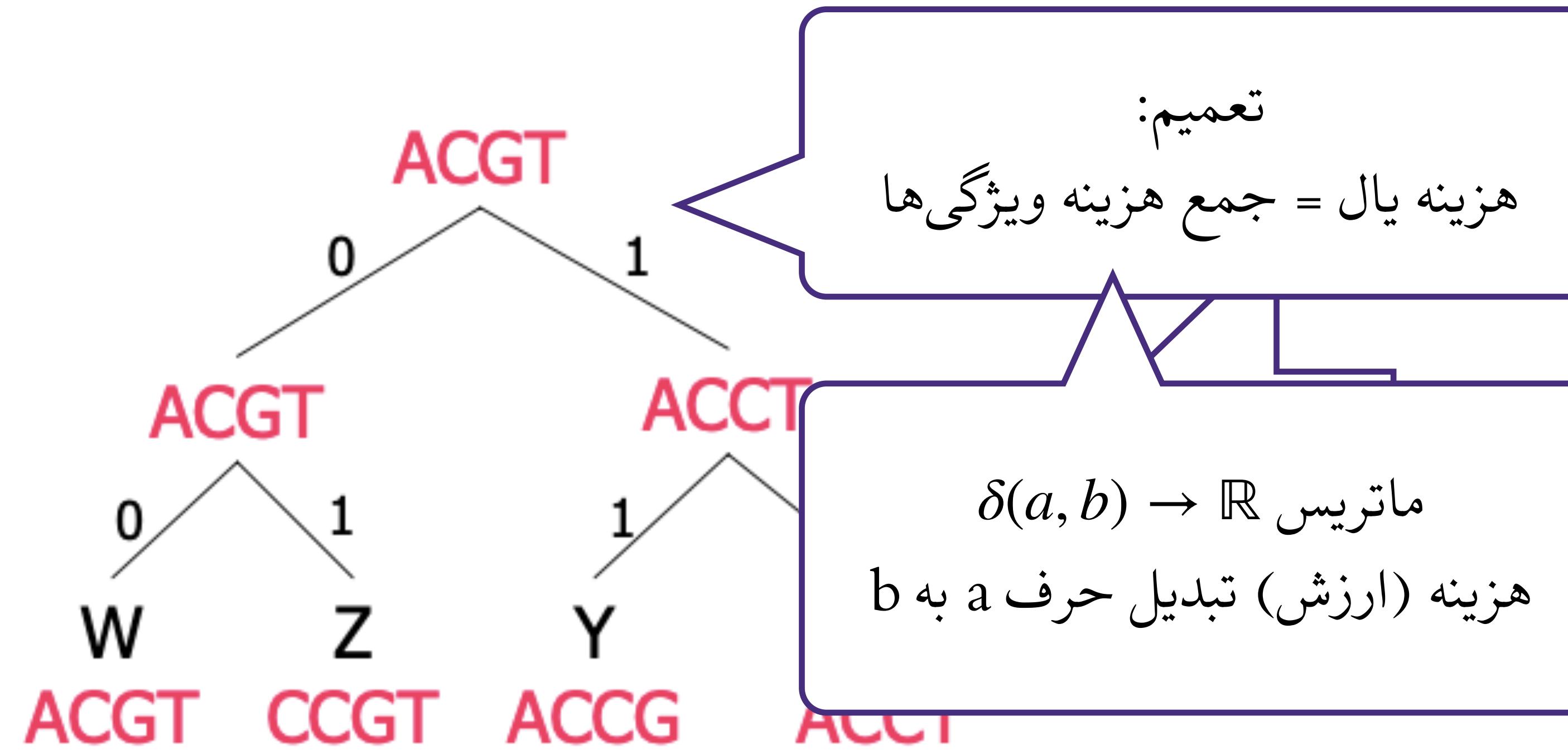


	1	2	3	4
W	A	C	G	T
X	A	C	C	T
Y	A	C	C	G
Z	C	C	G	T

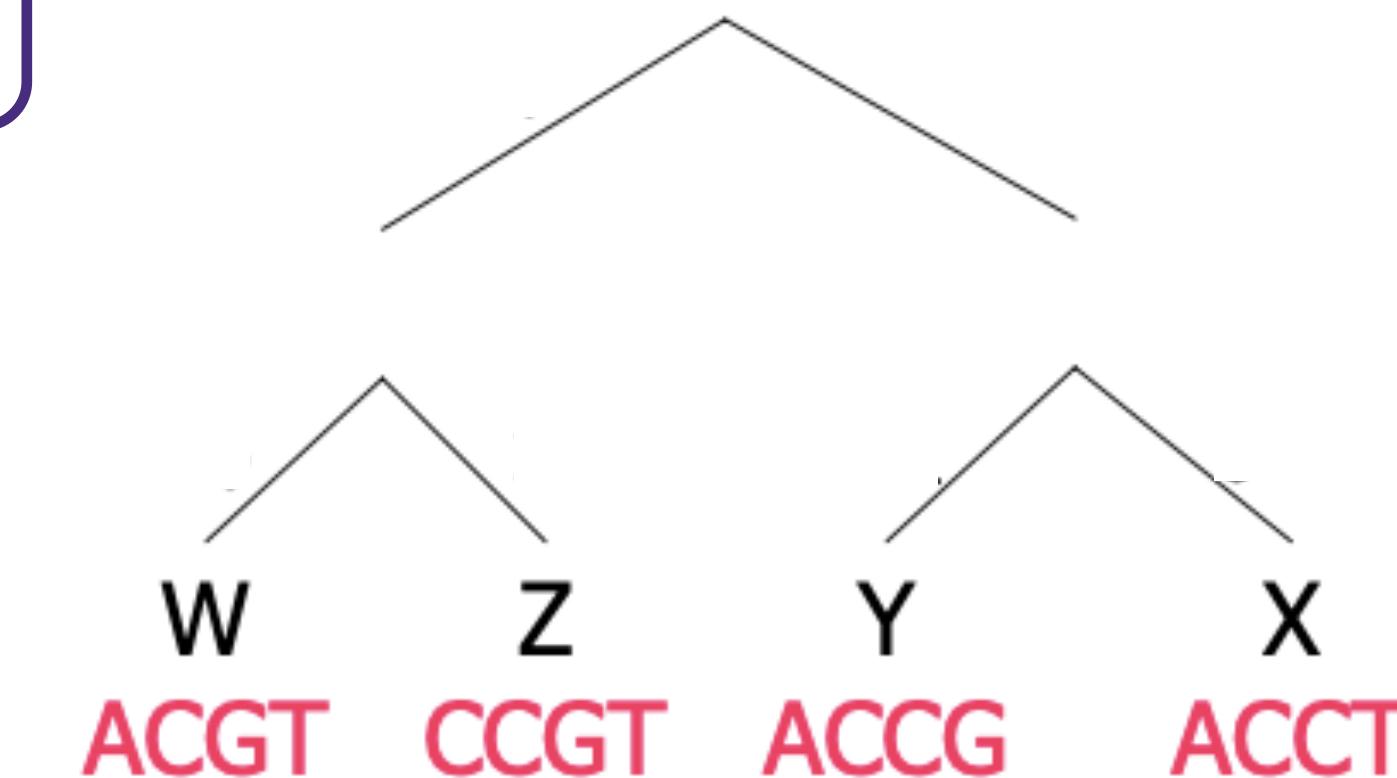
مسئله صرفه جویانه‌ترین درخت، با داشتن توپولوژی



مسئله صرفه جویانه‌ترین درخت، با داشتن توپولوژی

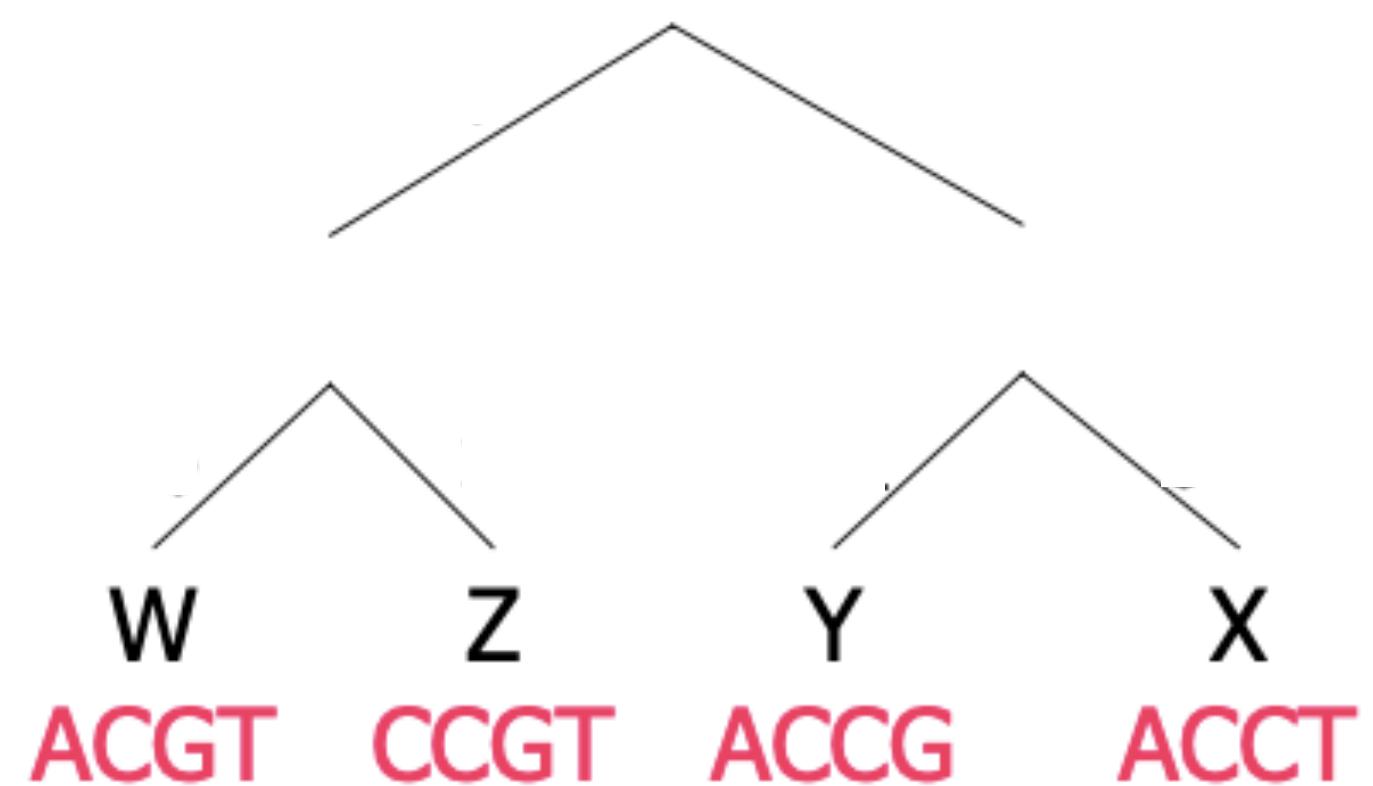


	1	2	3	4
W	A	C	G	T
X	A	C	C	T
Y	A	C	C	G
Z	C	C	G	T



صرفه‌جویانه‌ترین درخت، با داشتن توپولوژی: الگوریتم Sankoff

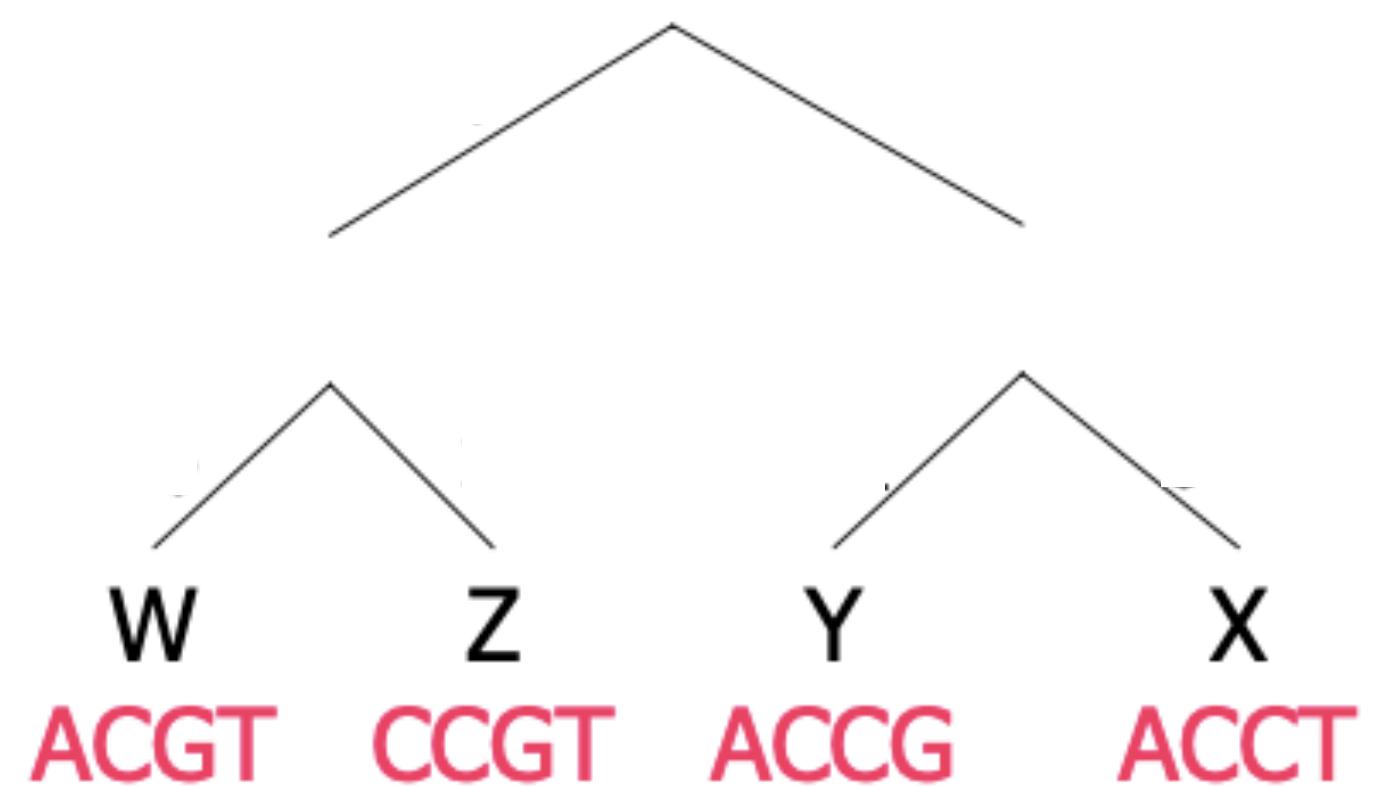
	1	2	3	4
W	A	C	G	T
X	A	C	C	T
Y	A	C	C	G
Z	C	C	G	T



صرفه‌جویانه‌ترین درخت، با داشتن توپولوژی: الگوریتم Sankoff

	1	2	3	4
W	A	C	G	T
X	A	C	C	T
Y	A	C	C	G
Z	C	C	G	T

- هر ویژگی جداگانه، برای ویژگی i :



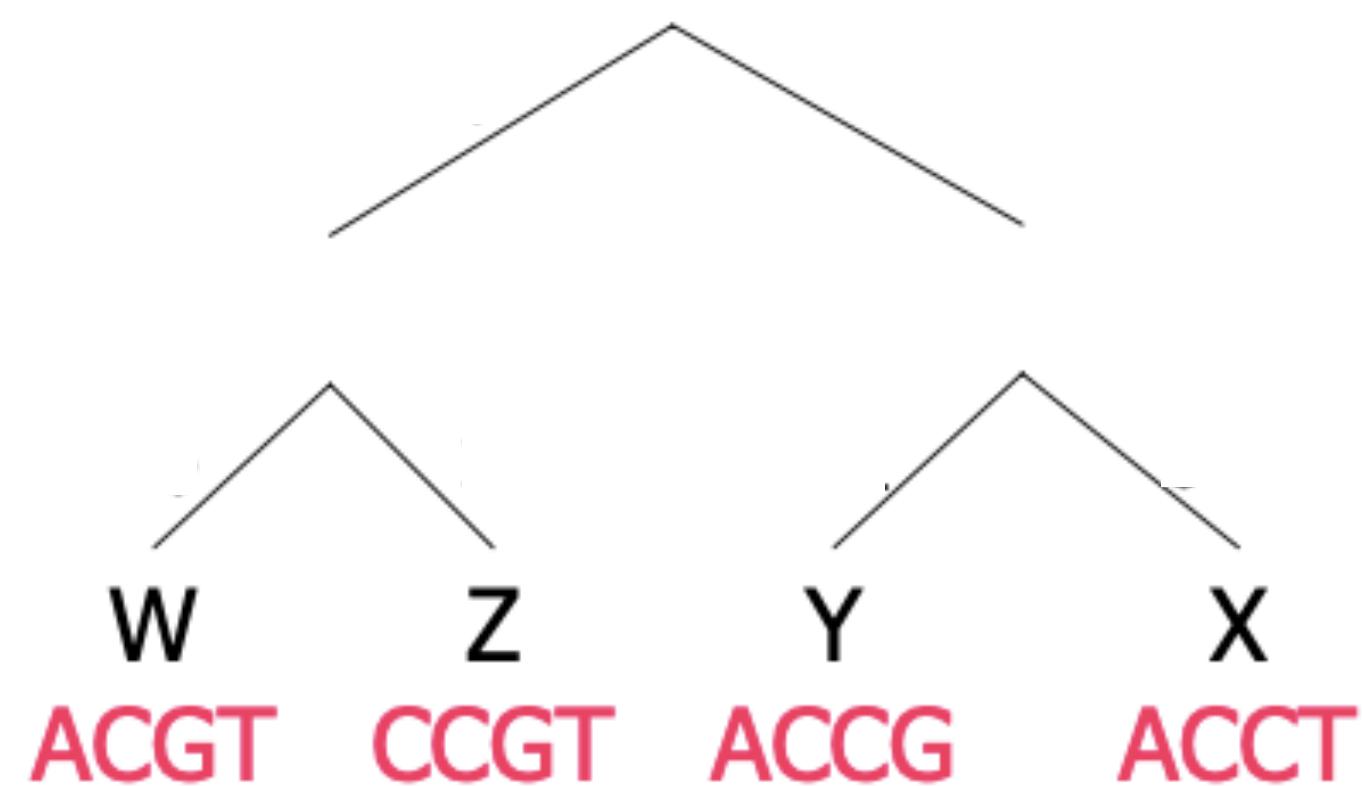
صرفه‌جویانه‌ترین درخت، با داشتن توپولوژی: الگوریتم Sankoff

	1	2	3	4
W	A	C	G	T
X	A	C	C	T
Y	A	C	C	G
Z	C	C	G	T

- هر ویژگی جداگانه، برای ویژگی i :

● تعریف $A[v, c]$: بیشترین امتیاز زیر درخت v ،

● به شرط انتساب c به راس v



صرفه‌جویانه‌ترین درخت، با داشتن توپولوژی: الگوریتم Sankoff

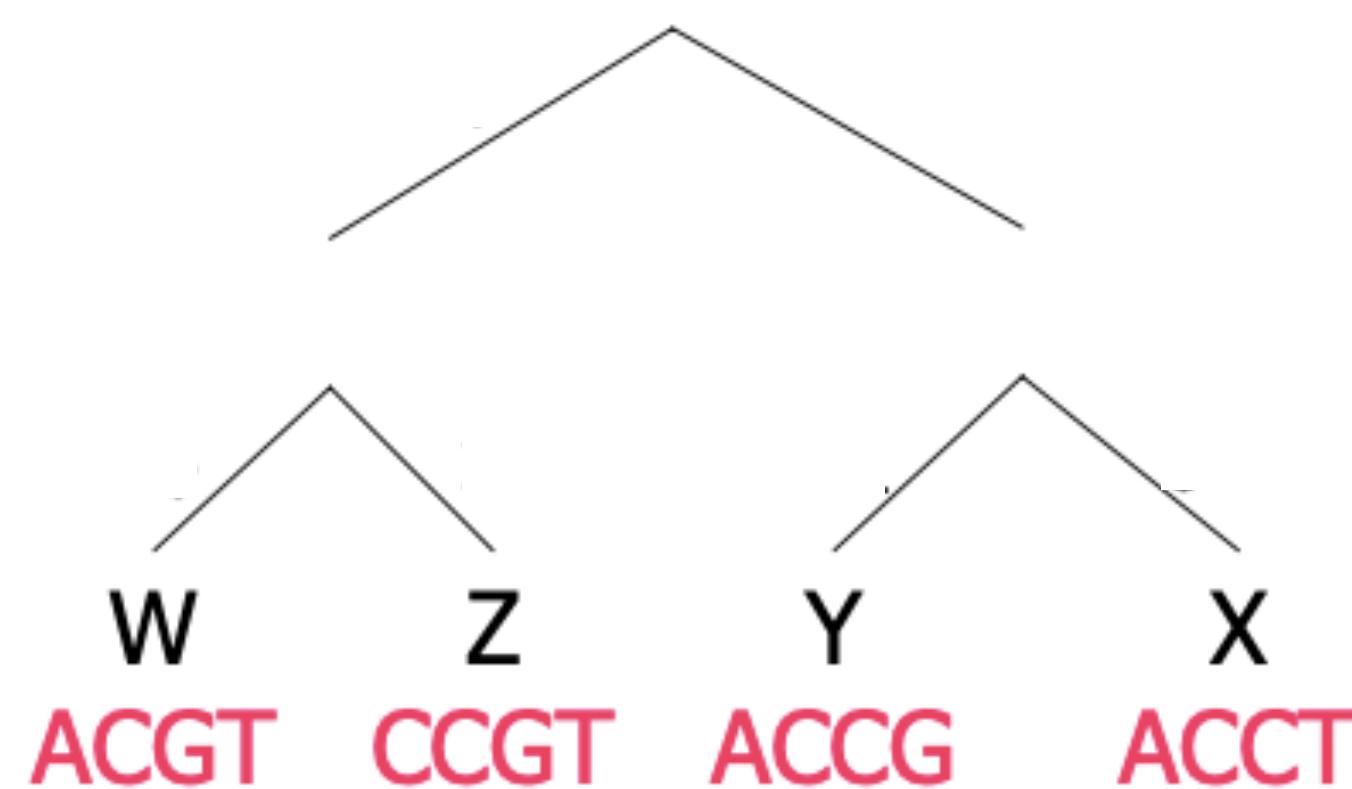
	1	2	3	4
W	A	C	G	T
X	A	C	C	T
Y	A	C	C	G
Z	C	C	G	T

- هر ویژگی جداگانه، برای ویژگی i :

● تعریف $A[v, c]$: بیشترین امتیاز زیر درخت v ،

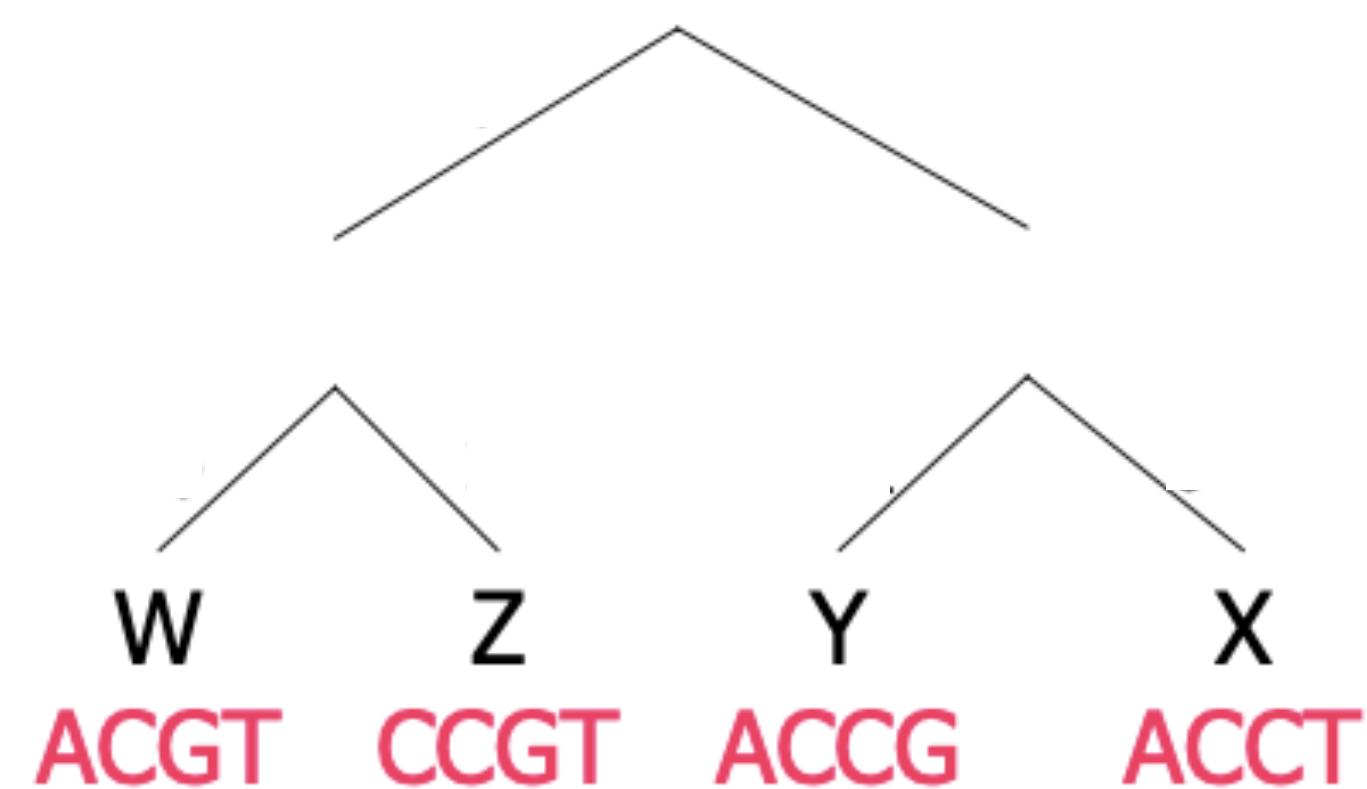
● به شرط انتساب c به راس v

$$A[v, c] = \sum_{u:v \rightarrow u} \max_{c' \in \Sigma} \delta(c, c') + A[u, c']$$



صرفه‌جویانه‌ترین درخت، با داشتن توپولوژی: الگوریتم Sankoff

	1	2	3	4
W	A	C	G	T
X	A	C	C	T
Y	A	C	C	G
Z	C	C	G	T



● رابطه بازگشتی $A[v, c] = \sum_{u:v \rightarrow u} \max_{c' \in \Sigma} \delta(c, c') + A[u, c']$

● حالت پایه: برای برگ v با ویژگی i برابر با a

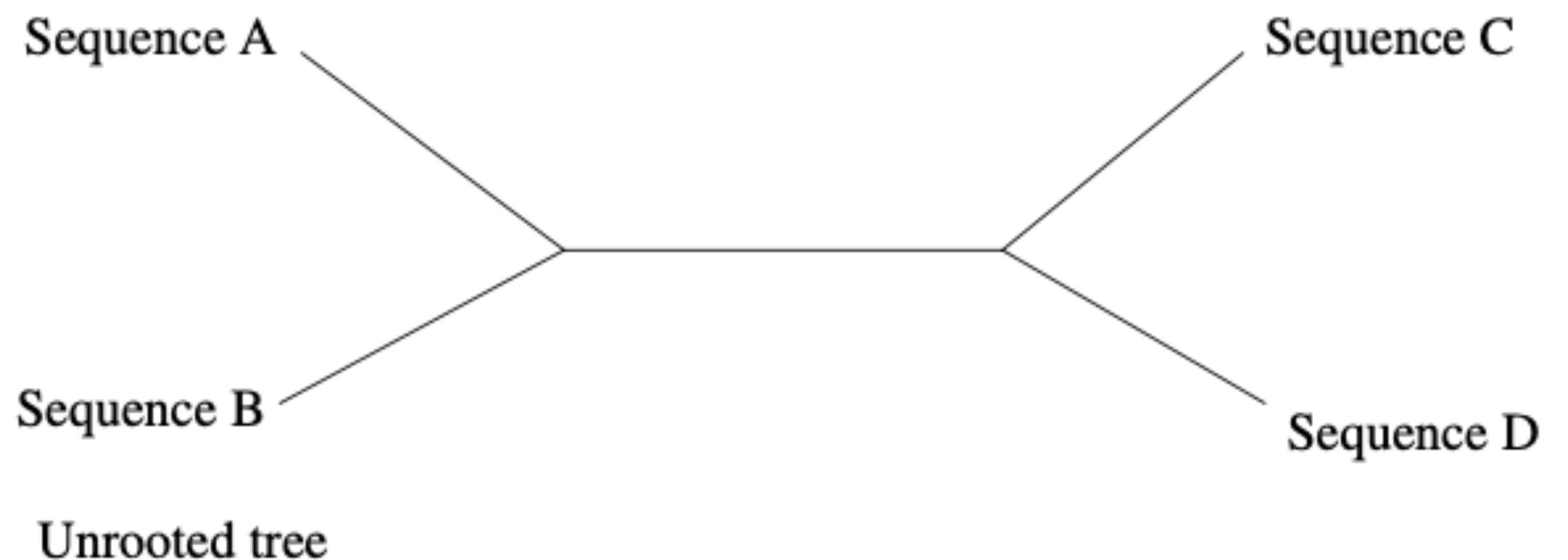
$$A[v, c] = \begin{cases} 0 & c = a \\ -\infty & \text{o.w.} \end{cases}$$

● هر ویژگی جداگانه، برای ویژگی i :

● تعریف $A[v, c]$: بیشترین امتیاز زیر درخت v ،

● به شرط انتساب c به راس v

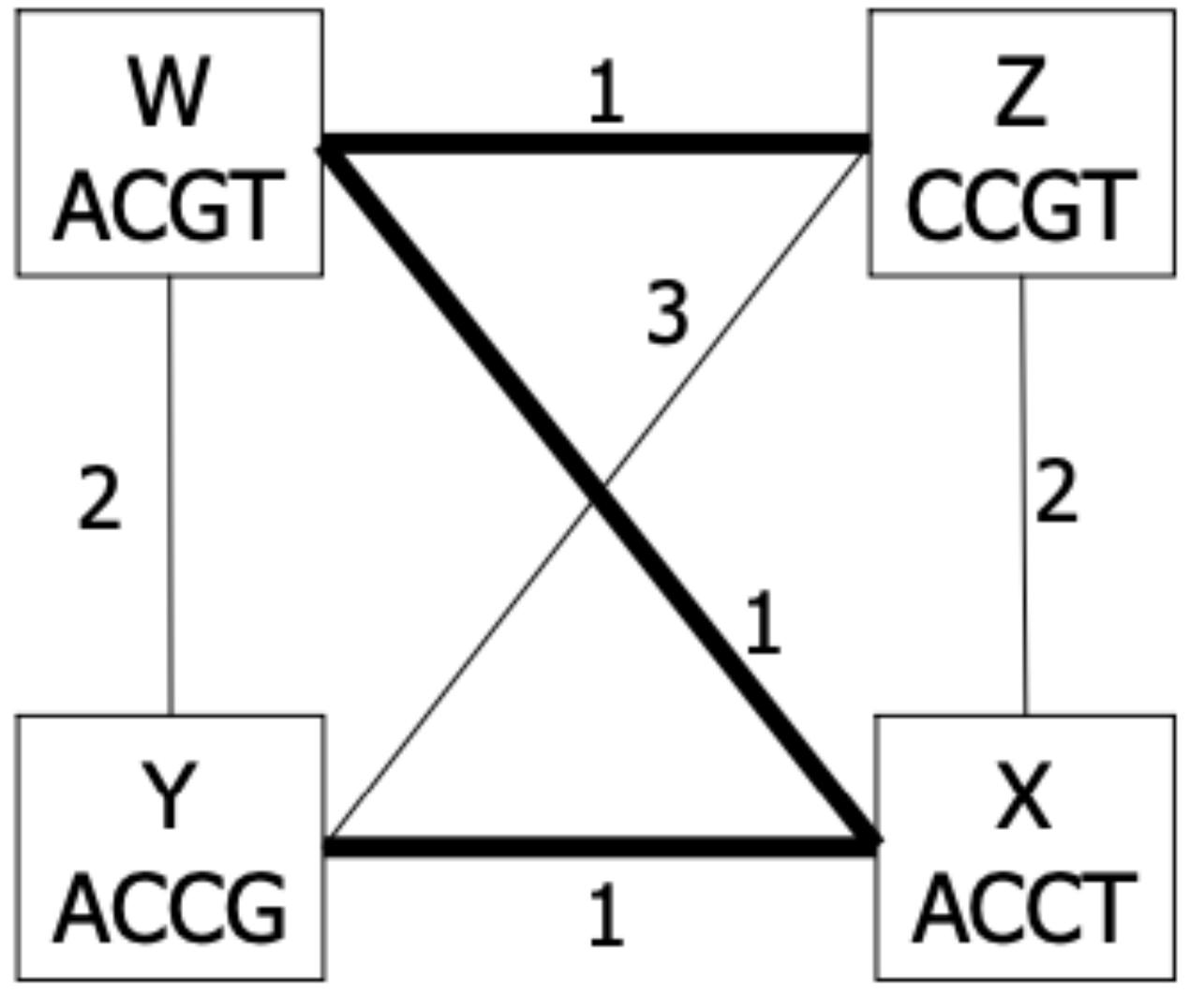
بهترین درخت



- بررسی همه درخت‌ها
- تعداد درخت‌ها؟
- تعداد یال‌های درخت با n برگ؟
- تعداد جاها برای اضافه کردن یک برگ جدید
- تعداد $= (2n - 1)!!$

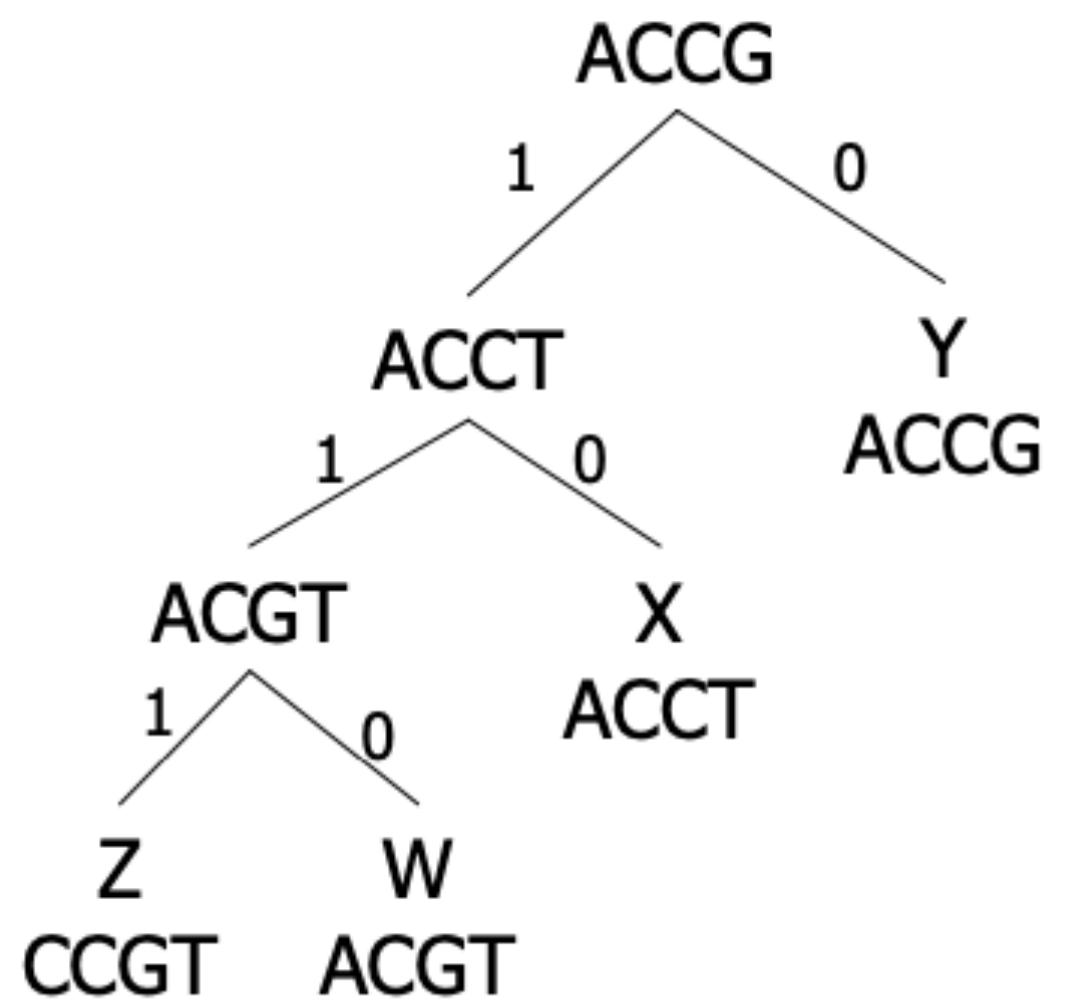
مسئله پاftن درخت تبارزایی صرفهجویانه NP_سخت است

الگوریتم تقریبی اول



$G(S)$

حذف برگ ==> راس میانی با برچسب غیربرگ

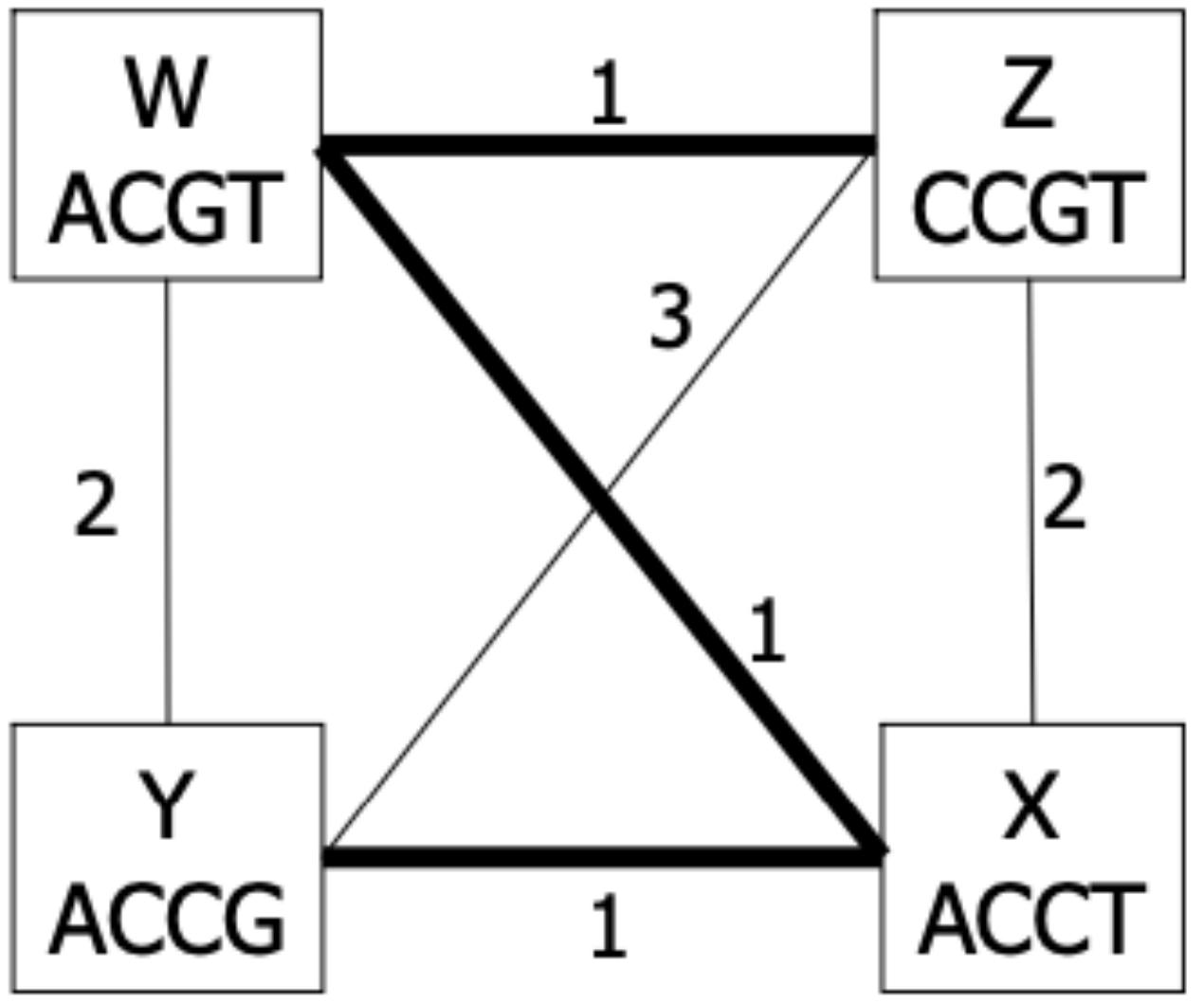


- الگوریتم:

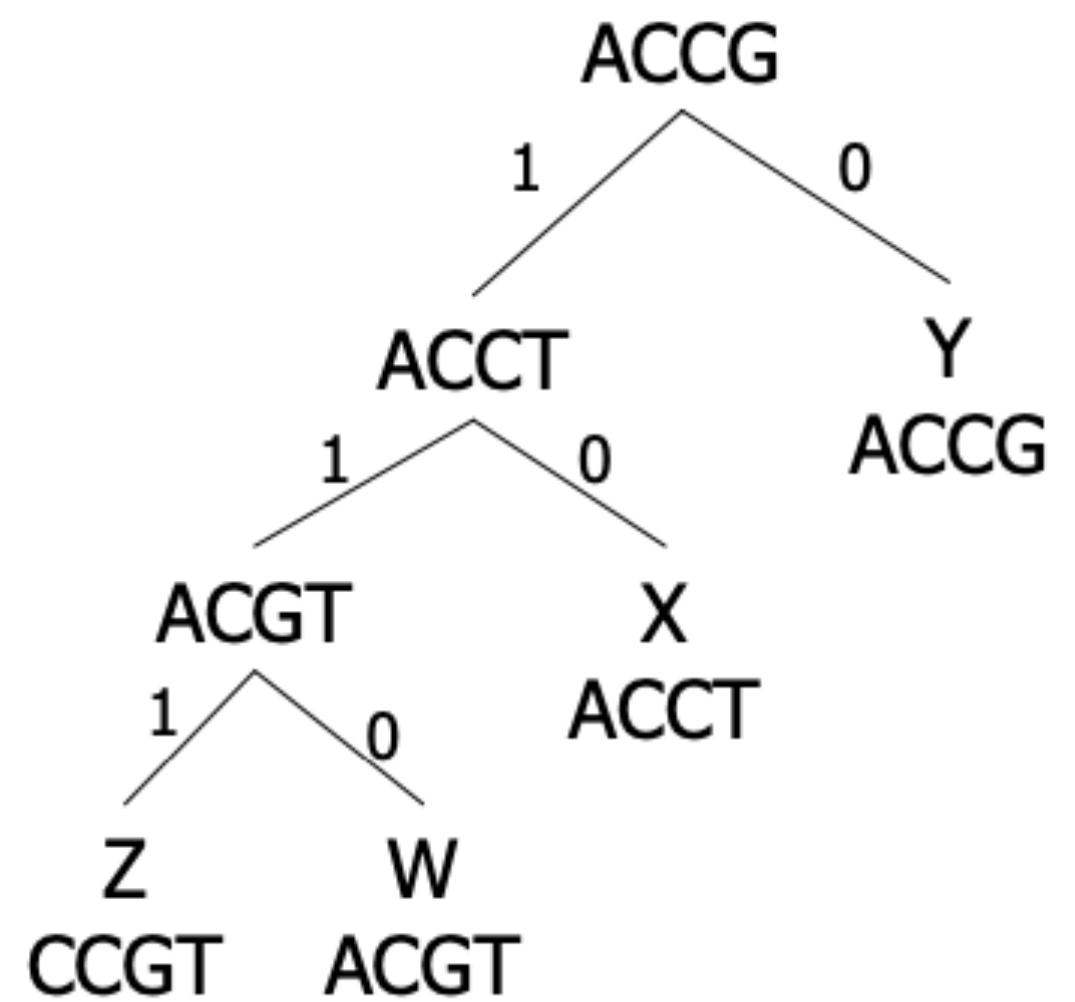
- گراف وزن دار بین هر دو گونه

- درخت فراگیر کمینه بین گونه ها

الگوریتم تقریبی اول



$G(S)$

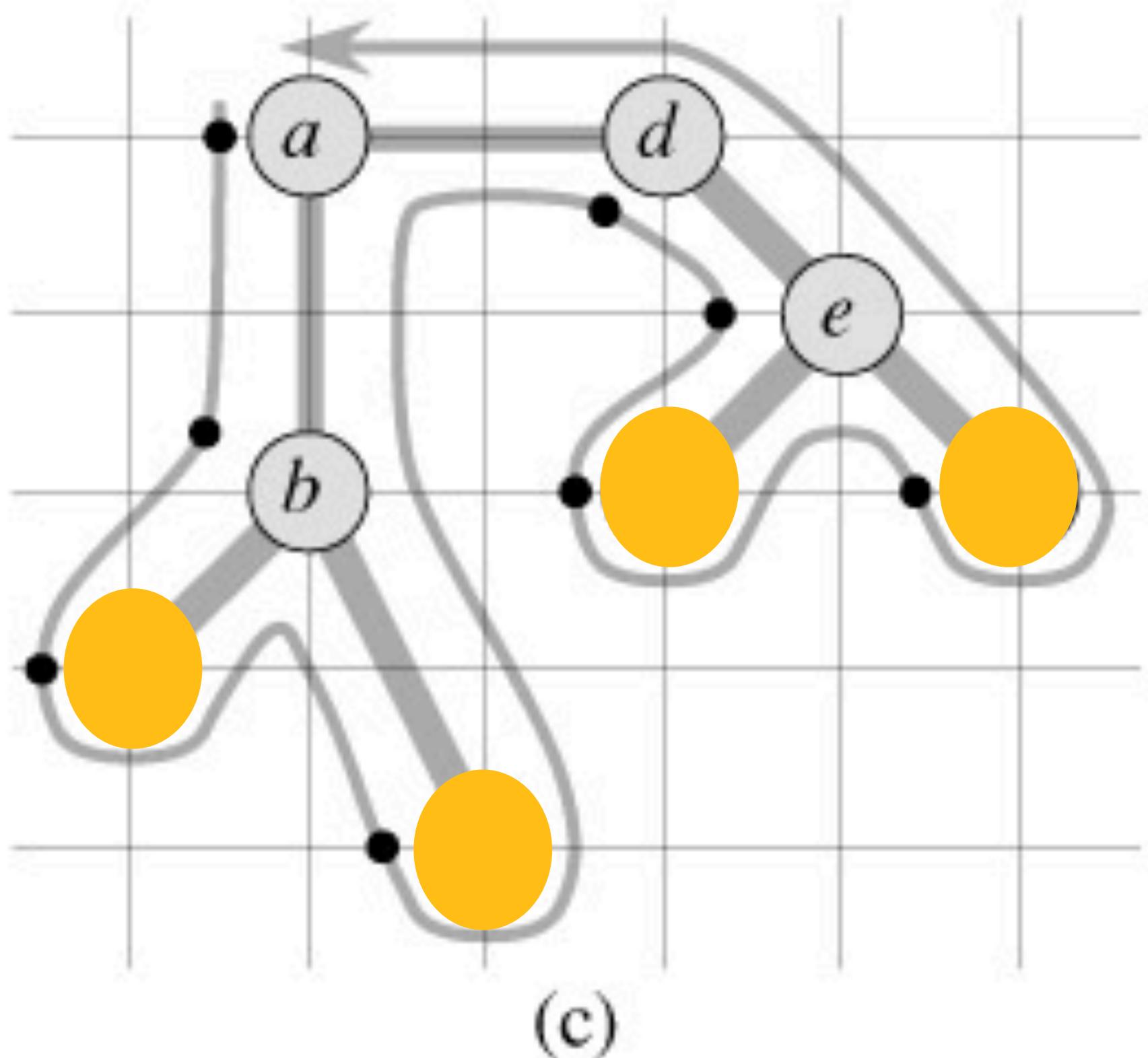


- حذف برگ \Rightarrow راس میانی با برحسب غیربرگ

- قضیه: ضریب تقریب الگوریتم ۲ است.

- الگوریتم:
- گراف وزن دار بین هر دو گونه
- درخت فراگیر کمینه بین گونه ها

قضیه: ضریب تقریب الگوریتم ۲ است



- درخت بهینه $= T^*$
- دور همیلتونی از روی درخت
- هزینه دور همیلتونی $= 2$ برابر هزینه T^*
- هزینه درخت ما $>=$ هزینه دور همیلتونی
- دور همیلتونی بعد از حذف یک یال، یک درخت فراگیر است. ما درخت کمینه را برداشتمیم.
- هزینه درخت ما $>= 2$ برابر هزینه T^*

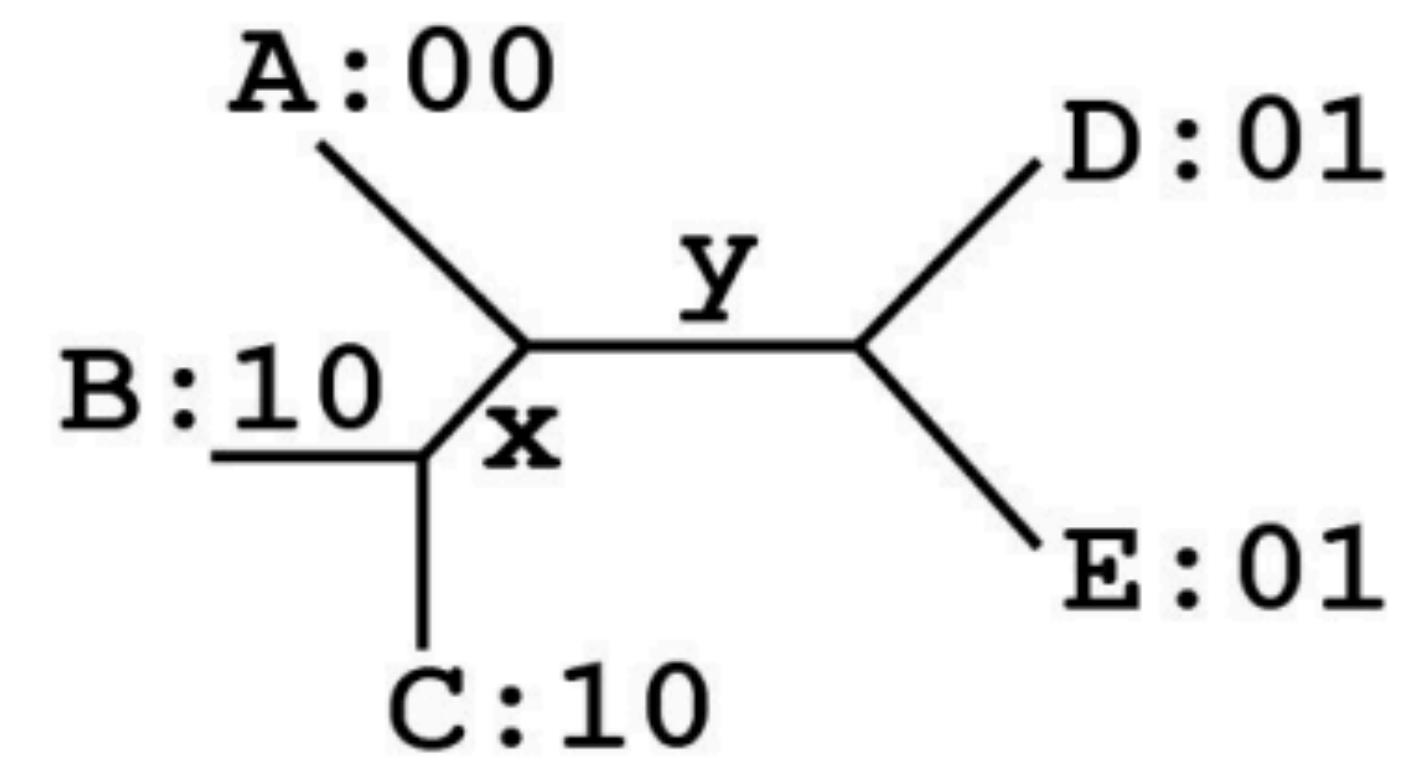
الگوريتم تقريري دوم (با ضريرب ۱/۵)

درخت بی نقص

Two Binary Characters

	x	y
A:	0	0
B:	1	0
C:	1	0
D:	0	1
E:	0	1

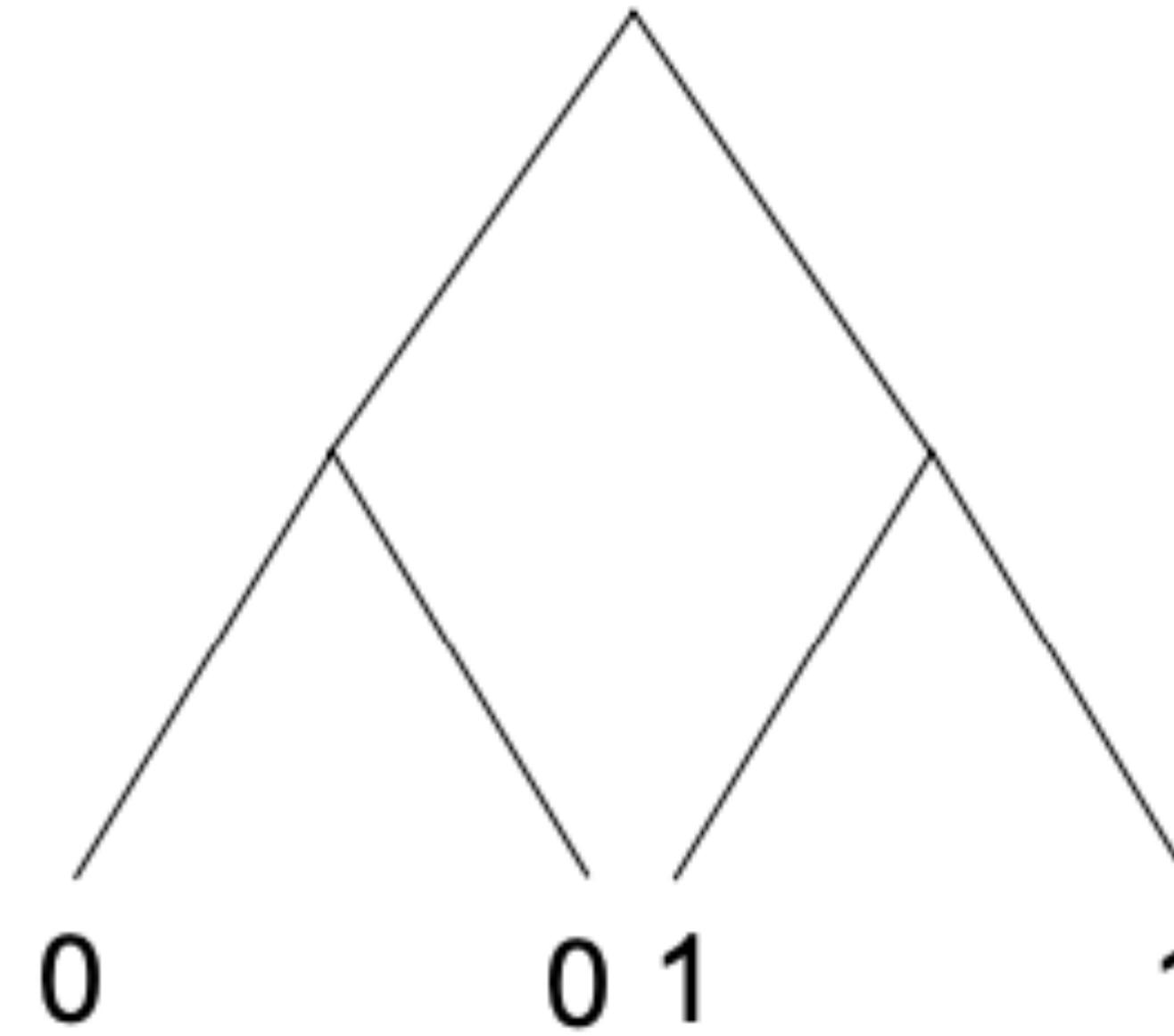
Perfect Phylogeny explaining x & y



x & y mutate on indicated branches

سازگاری درخت با یک ویژگی

فرض ● ISA



سازگاری ویژگی

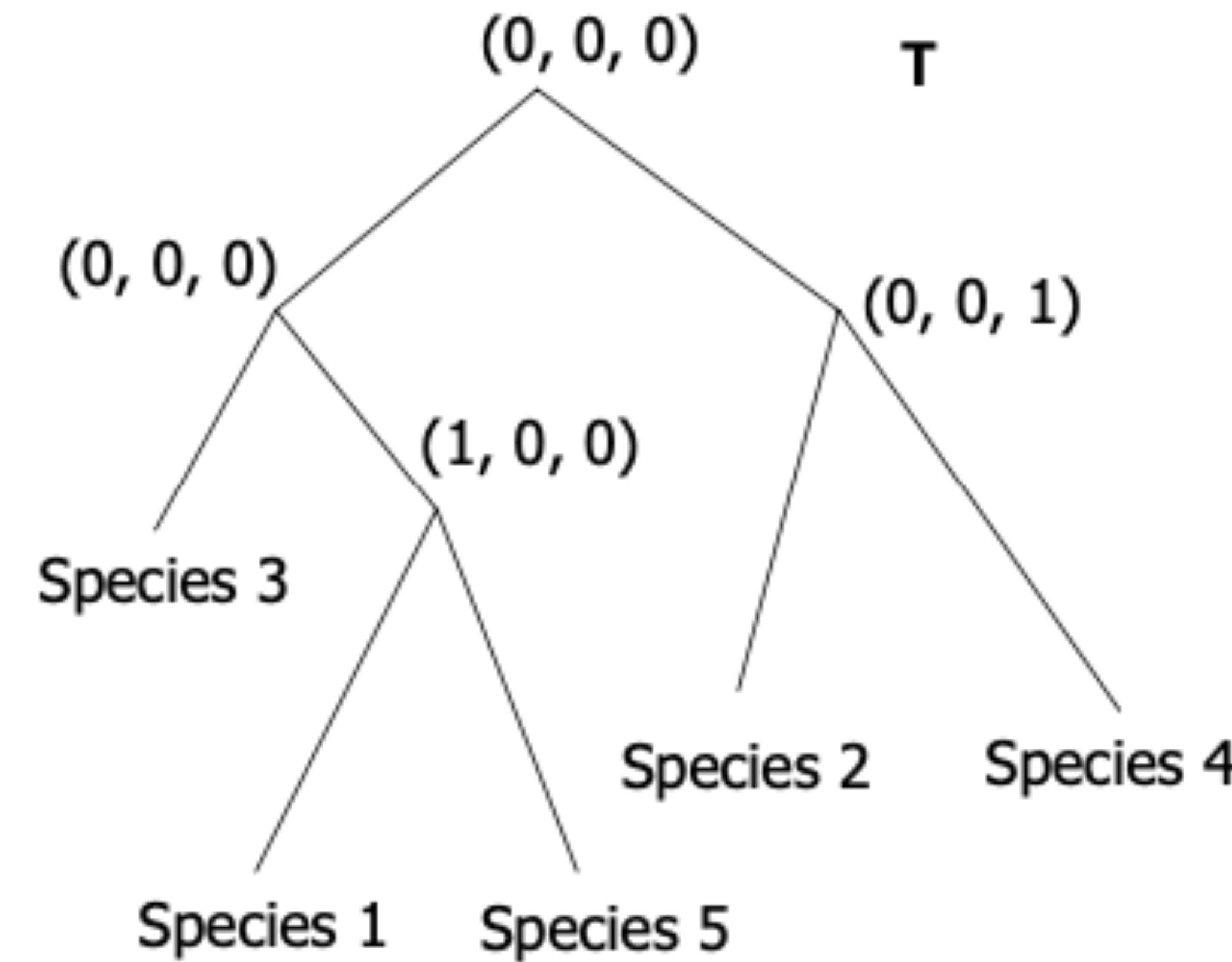
ناسازگاری ویژگی

مسئله درخت تبارزایی بی نقص

- ورودی: ماتریس ویژگی‌ها
- خروجی:
- اگر درخت T هست که با همه ویژگی‌ها سازگار است
-> یک درخت
- وگرنه «هیچ درختی نیست»

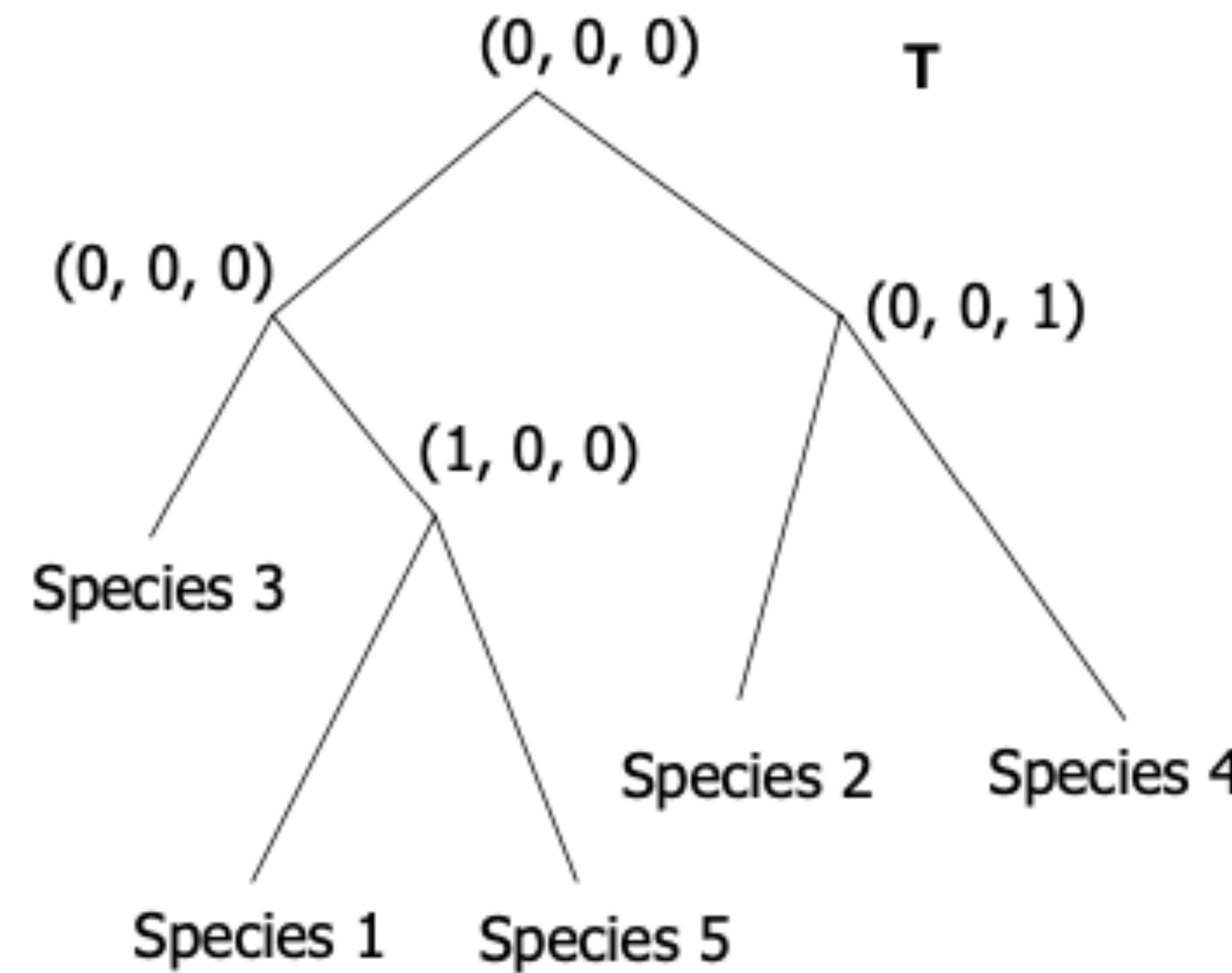
مسئله درخت تبارزایی بی نقص (مثال)

M	X ₁	X ₂	X ₃
Species 1	1	1	0
Species 2	0	0	1
Species 3	0	0	0
Species 4	0	0	1
Species 5	1	0	0



یال جداکننده

M	X ₁	X ₂	X ₃
Species 1	1	1	0
Species 2	0	0	1
Species 3	0	0	0
Species 4	0	0	1
Species 5	1	0	0



سازگاری ۲ به ۲ <=> وجود درخت بی نقص

● فرض: برای هر ویژگی تعداد اها $=$ تعداد های

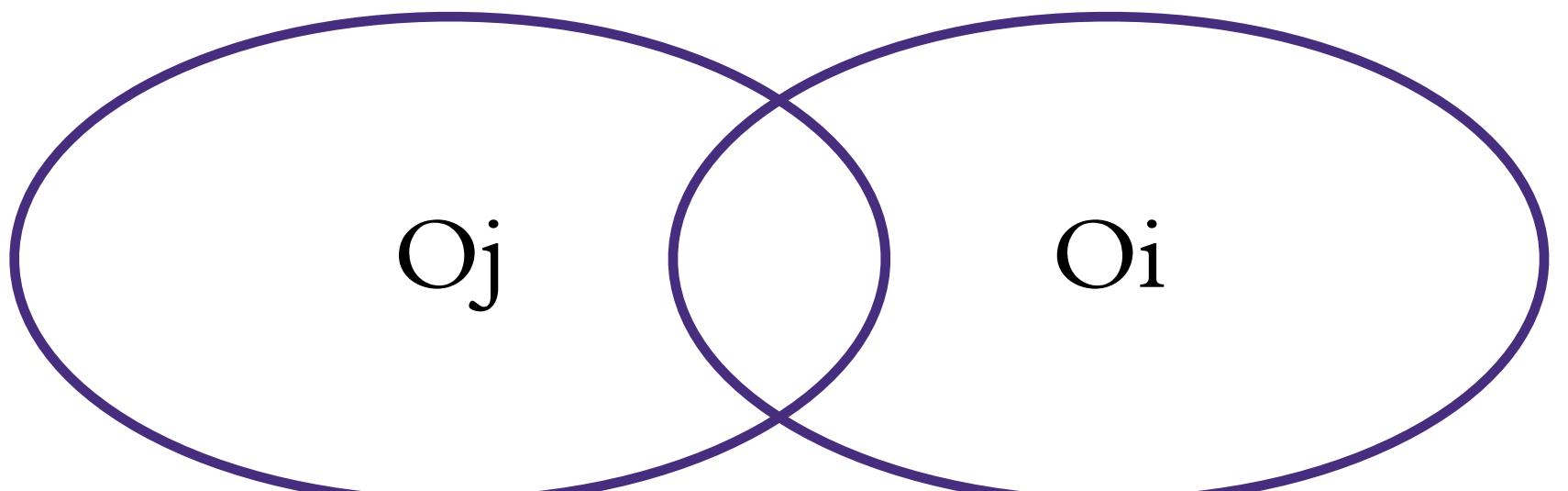
● O_i : گونه های با ویژگی $i = 1$

● دو ویژگی i و j سازگارند اگر یکی از حالت های زیر باشد

$$O_i \cap O_j = \emptyset \text{ یا } O_j \subseteq O_i \subseteq O_j$$

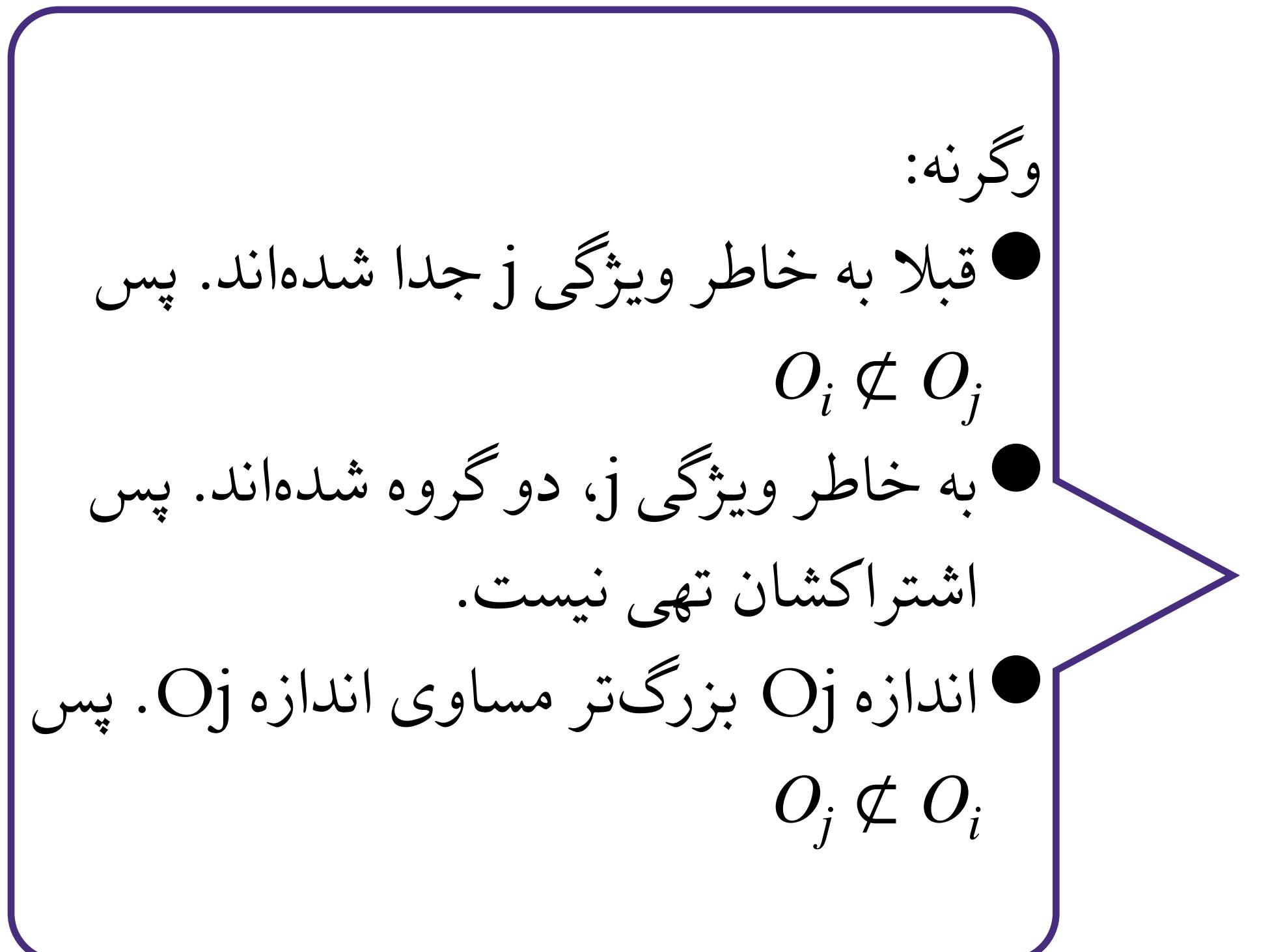
● قضیه: درخت فیلوژنی برای ماتریس M وجود دارد اگر و فقط اگر هر دو ویژگی M سازگار باشند.

M	i	j
	1	0
	0	1
	1	1
	0	0



● درخت $=>$ سازگاری دو به دو

قضیه: درخت فیلوژنی برای ماتریس M وجود دارد اگر و فقط اگر دو ویژگی M سازگار باشند.

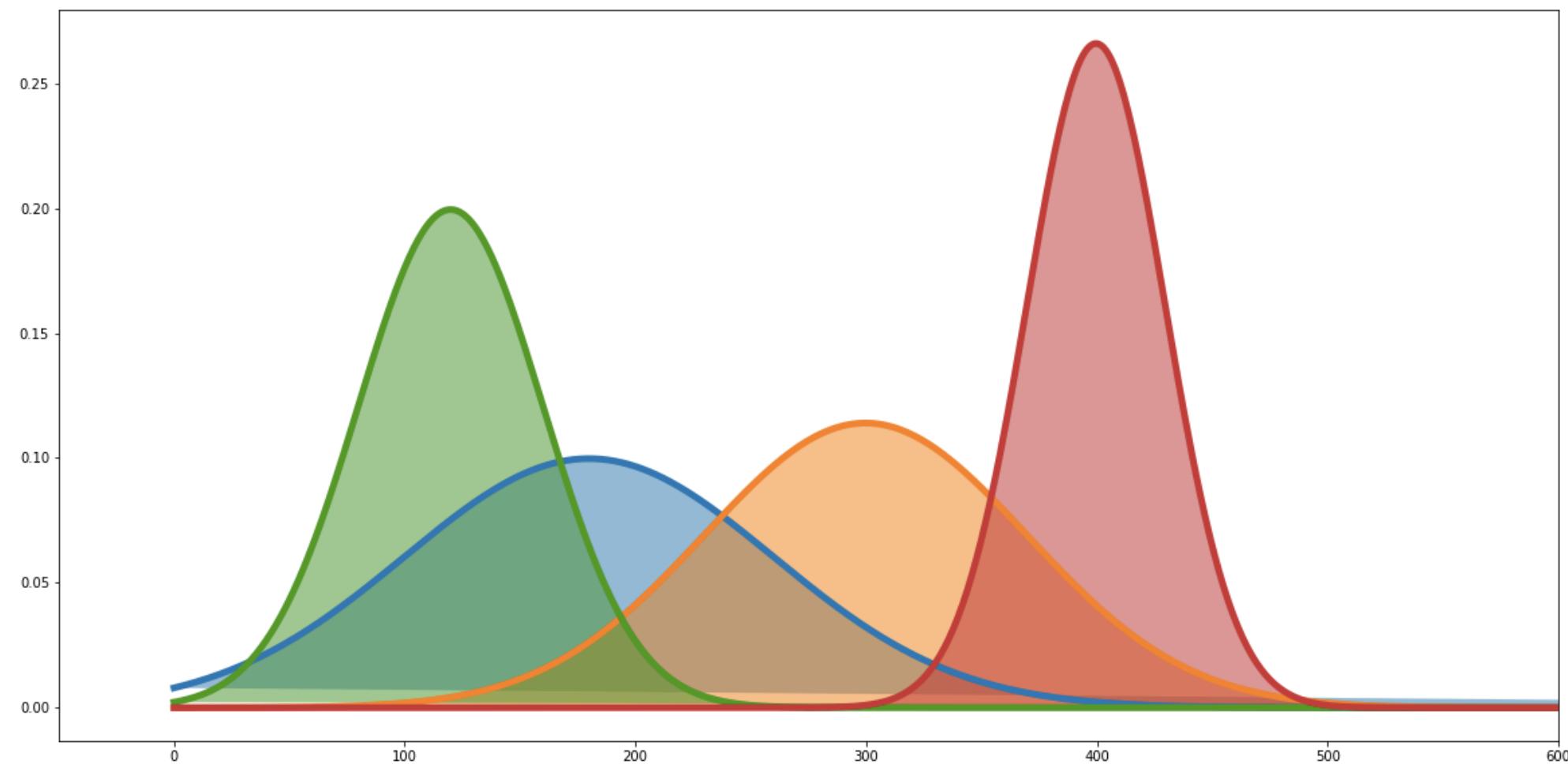


- سازگاری \Rightarrow درخت
- مرتب بر حسب $|O_i|$
- // یک درخت، در هر برگ تعدادی گونه، در هر مرحله گونه‌های یک برگ را به دو گروه تقسیم می‌کنیم.
- در مرحله i :
- همه گونه‌های O_i در یک برگ هستند
- آن برگ را دو قسمت کن، O_i ‌ها در یک زیردرخت و بقیه در یک زیردرخت

پیدا کردن درخت بی نقص با کمترین تغییر

- تغییر: تغییر 0 و 1 در ماتریس
- تغییر: حذف ویژگی
- $= < \rightarrow$ هر دو سخت!

برآورد درستنمايی پيشينه



برآورد درستنمایی بیشینه

مثال:

یک سکه احتمال شیر p , احتمال خط $1-p$

تعدادی آزمایش

? = p

روش برآورد درستنمایی بیشینه: کدام p بوده باشد با احتمال بیشتری همین نتیجه آزمایش را تولید می‌کند؟

مثال:

خط $p - (1-p)$ شیر

مثال: $p \in [0,1]$

مثال:

احتمال شیر آمدن

کلی:

یک مدل احتمالاتی $P(\theta)$ با پارامترهای $\Theta \in \Theta$ به هر حالت ورودی یک احتمال نسبت می‌دهد.

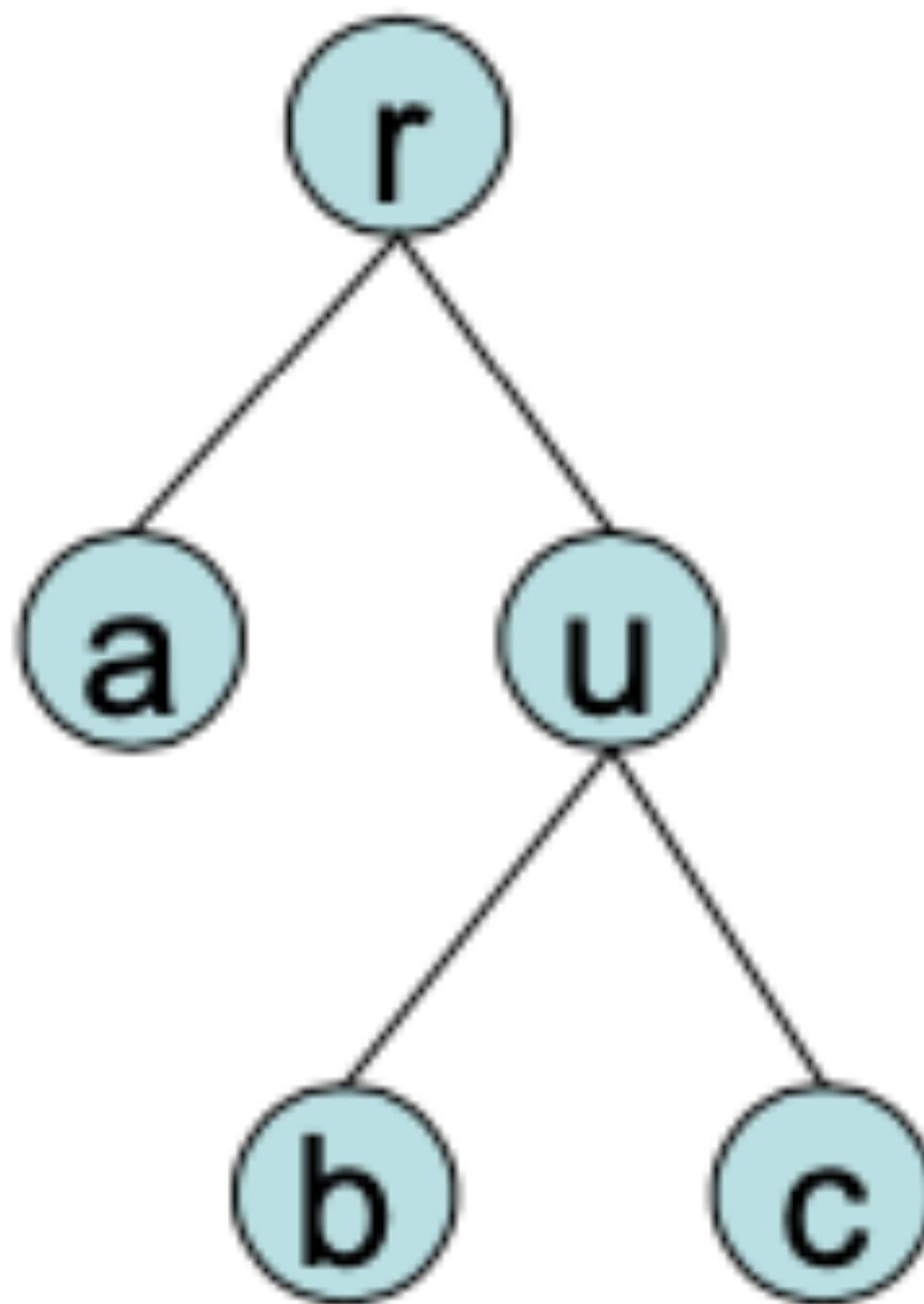
$$\max_{\theta \in \Theta} P(D | \theta)$$

کدام مدل (کدام θ) با احتمال بیشتری این ورودی را تولید می‌کند؟

تعریف: $L(\theta) := P(D | \theta)$

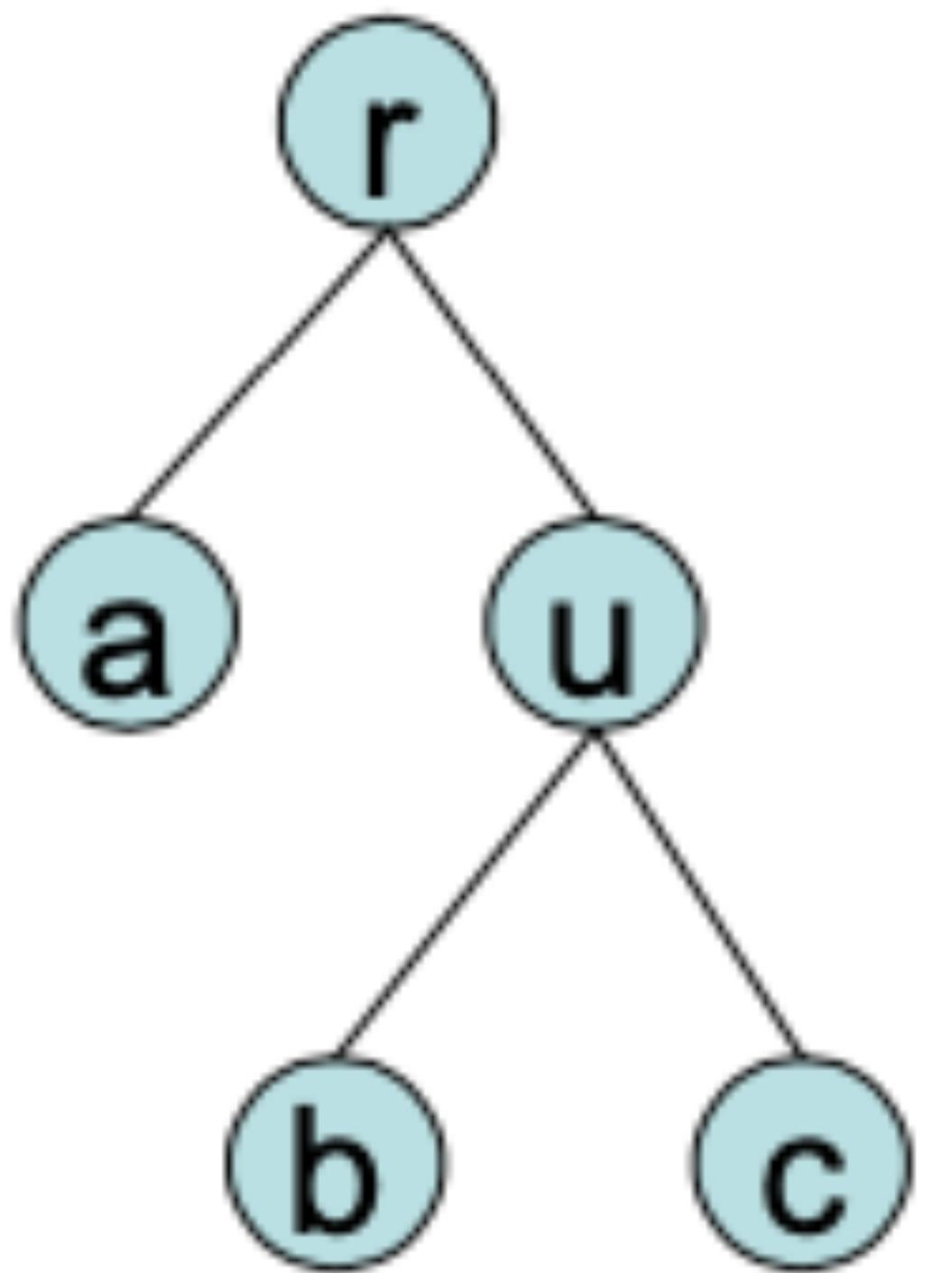
Model Cavender-Felsenstein

T



- ویژگی ها ۰ یا ۱ هستند.
- ویژگی ها مستقل تغییر می کنند.
- هر مدل: $T = (T, \{p_e \mid e \in T\})$
- هر یال e ، احتمال تغییر p_e
- احتمال تغییر هر ویژگی روی یال e
- احتمال هر حالت هر ویژگی در ریشه = $1/2$

T

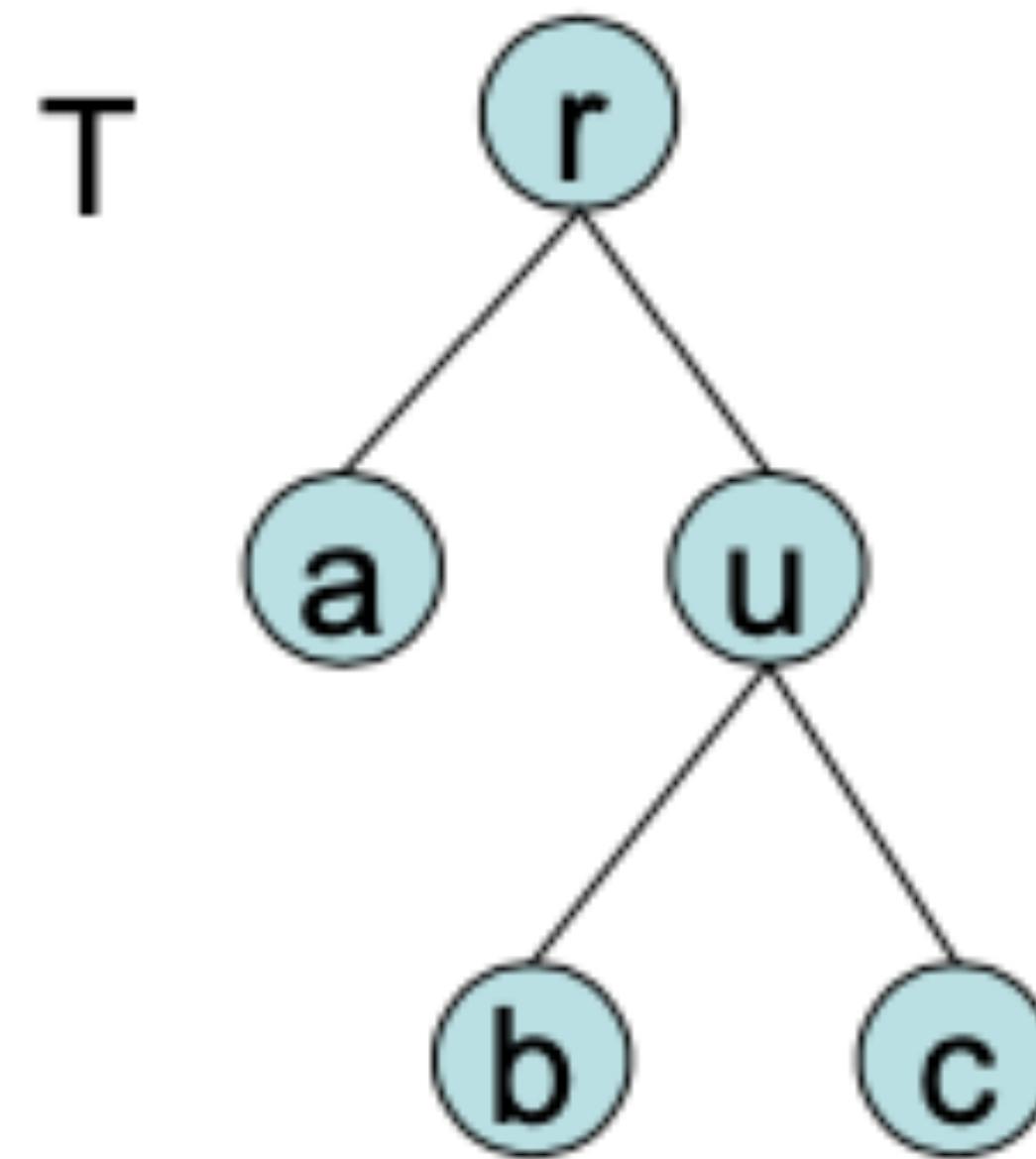


احتمال مشاهده
iu را روی راسها

اگر رشته ویژگی‌ها روی هر راس را داشته باشیم:

$$\begin{aligned} & Pr(u = i_u \forall u \in T | T) \\ &= Pr(r = i_r) \prod_{(u,v) \in T} [\delta(i_u, i_v)(1 - p_{(u,v)}) + (1 - \delta(i_u, i_v))p_{(u,v)}] \end{aligned}$$

اگر رشته ویژگی‌ها روی راس‌های میانی را نداشته باشیم



احتمال مشاهده
ها روی برگ‌ها

$$Pr(u = i_u \forall u \in L(T) | T) = \sum_{i_v \in \{0,1\} \forall v \in I(T)} Pr(u = i_u \forall u \in T | T)$$

همه حالت‌های
راس‌های میانی

مسئله بازسازی درخت با درست‌نمایی بیشینه با داشتن درخت:

$$\max_{p_e} P(i_u \forall u \in L(T) | \{T, p_e\})$$

مسئله بازسازی درخت با درست‌نمایی بیشینه:

$$\max_{p_e, T} P(i_u \forall u \in L(T) | \{T, p_e\})$$

محاسبه درستنمایی

● ورودی: ماتریس $M[v,i]$ (ویژگی i برای برگ v)

● تعریف: $L_i(v,s) =$

● بیشینه احتمال مشاهده ویژگی i روی برگ‌های زیردرخت v ، به شرط اینکه ویژگی i روی راس v برابر با s باشد.

● جواب: $\prod_i \left(\frac{1}{2}L_i(r,0) + \frac{1}{2}L_i(r,1) \right)$

● حالت پایه: برای برگ v : $L_i(v,s) = 1_{s=M[v,i]}$

● رابطه بازگشتی: برای راس میانی u با فرزندان v و w :

$$L_i(u,s) = \left[\sum_{y \in \{0,1\}} L_i(v,y) Pr(v_i = y | u_i = s) \right] \left[\sum_{x \in \{0,1\}} L_i(w,x) Pr(w_i = x | u_i = s) \right]$$

اگر رشته ویژگی‌ها روی هر راس را داشته باشیم:

مسئله بازسازی درخت با درست‌نمایی بیشینه:

$$\max_{p_e, T} P(i_u \forall u \in L(T) | \{T, p_e\})$$

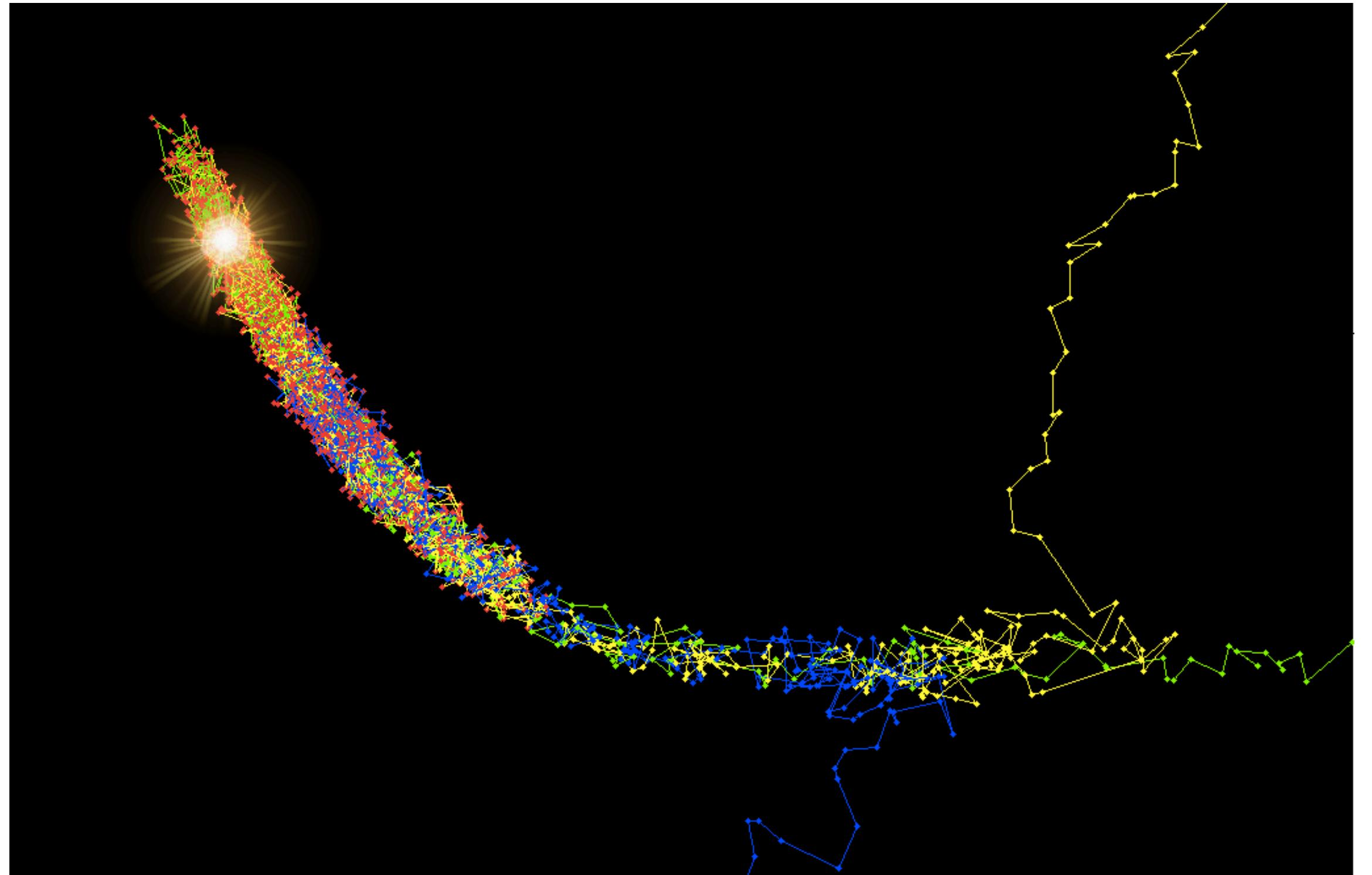
● الگوریتم‌های ابتکاری

● هر دفعه

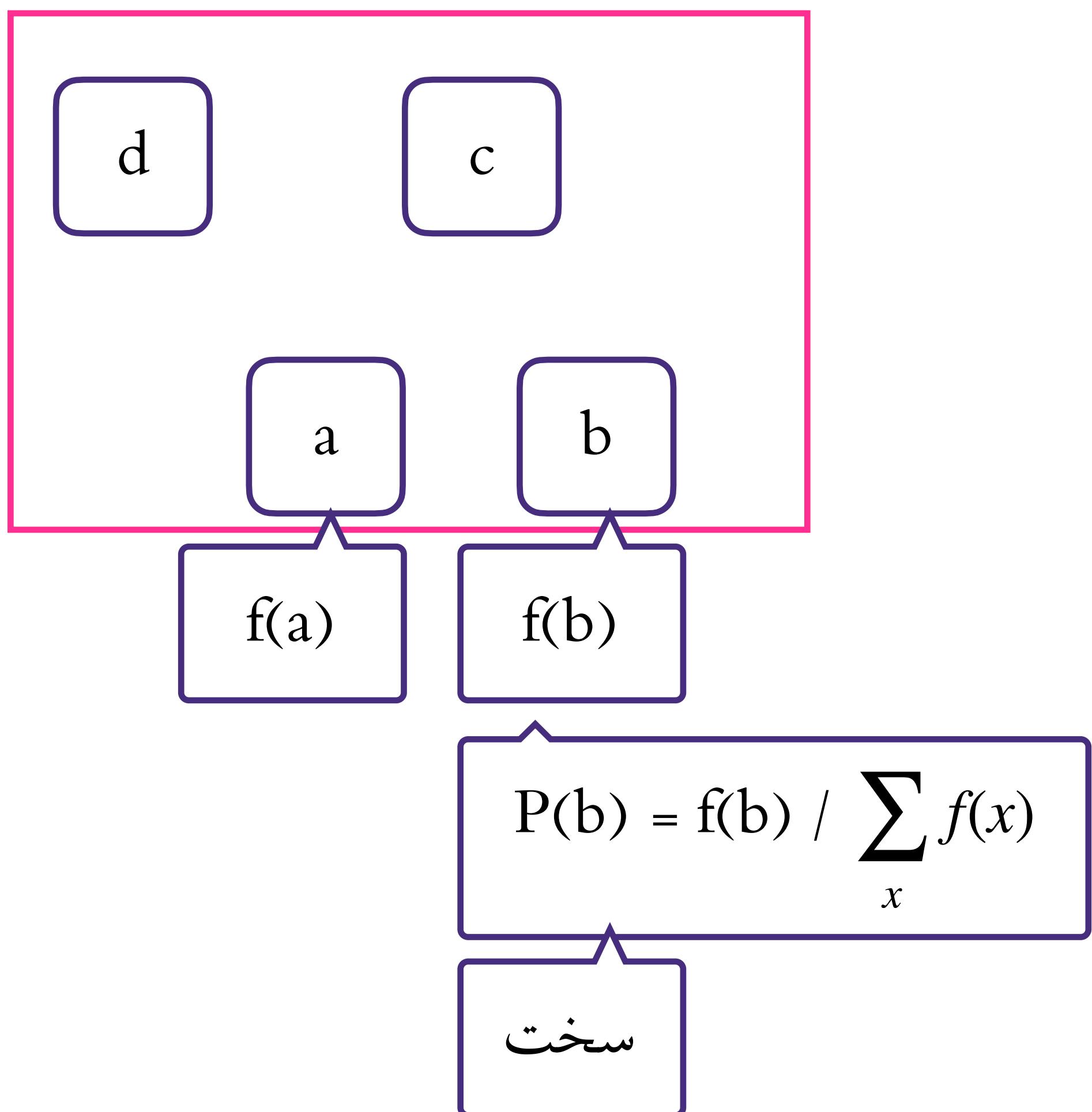
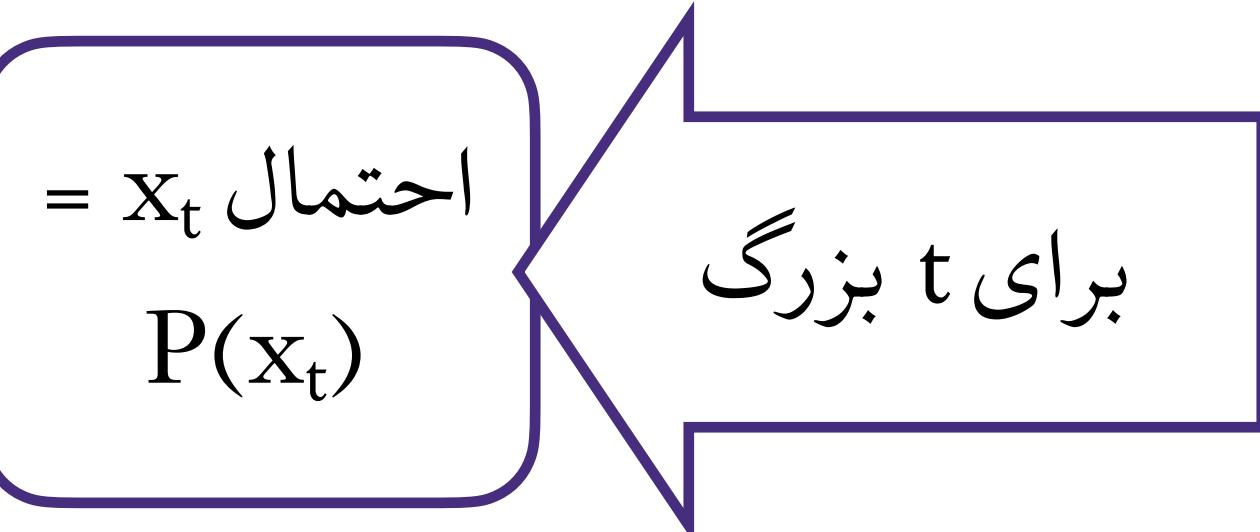
● یک تغییر روی درخت و روی pe‌ها،

● مشاهده L

MCMC



MCMC



- یک نقطه ابتدایی x_0
- هر دفعه
- یک همسایه از توزیع $g(x' | x_t)$ انتخاب کن
- با احتمال تغییر را بپذیر

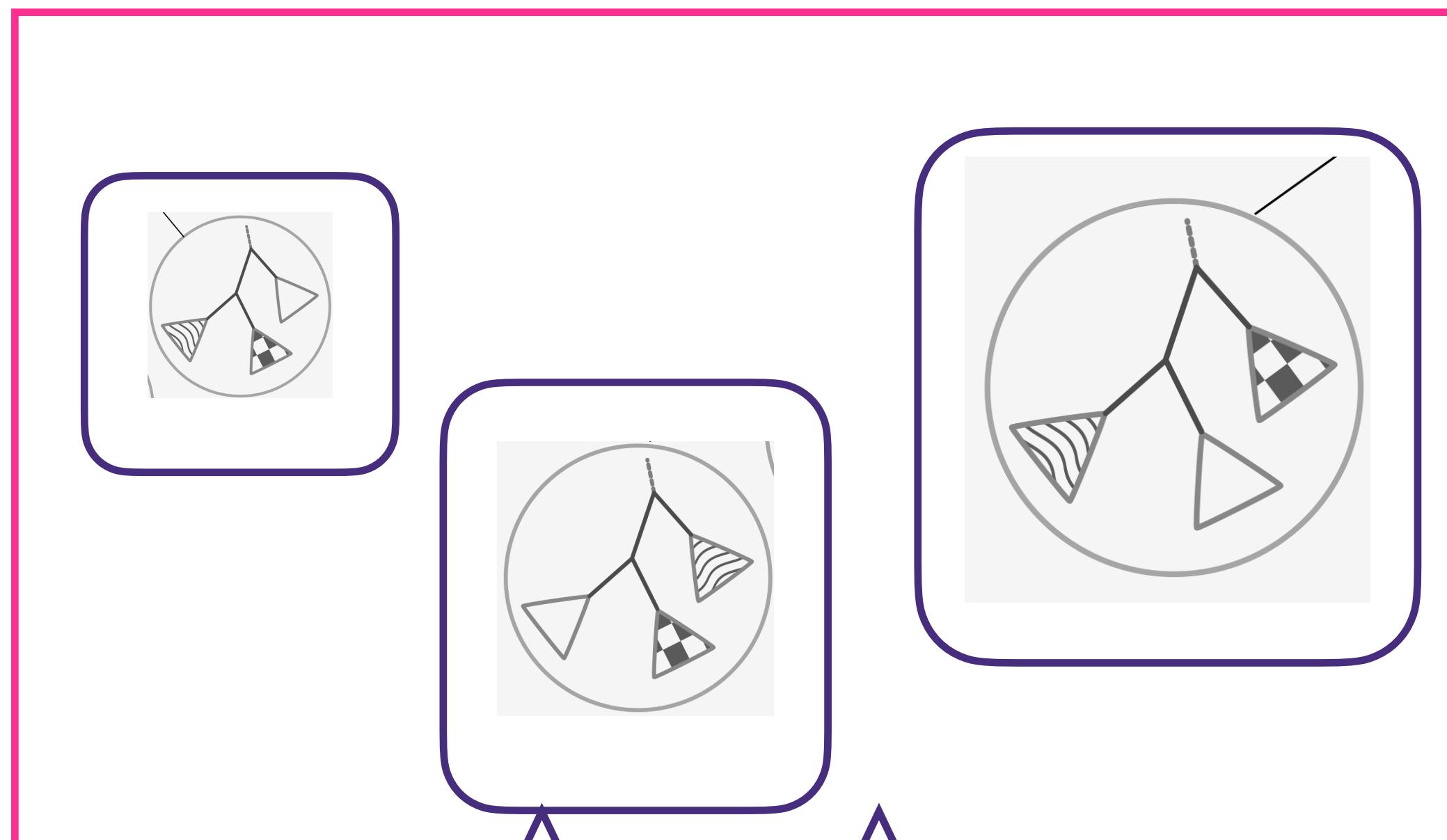
$$A(x', x_t) = \min \left(1, \frac{P(x')}{P(x_t)} \frac{g(x_t | x')}{g(x' | x_t)} \right)$$

کافی است نسبت
احتمال را محاسبه
کنیم

MCMC برای درخت

احتمال
 $P(x_t)$

برای t بزرگ



$L(a)$

$L(b)$

$$P(b) = f(b) / \sum_x f(x)$$

یک زیردرخت را بگیریم
جای دیگر بچسبانیم

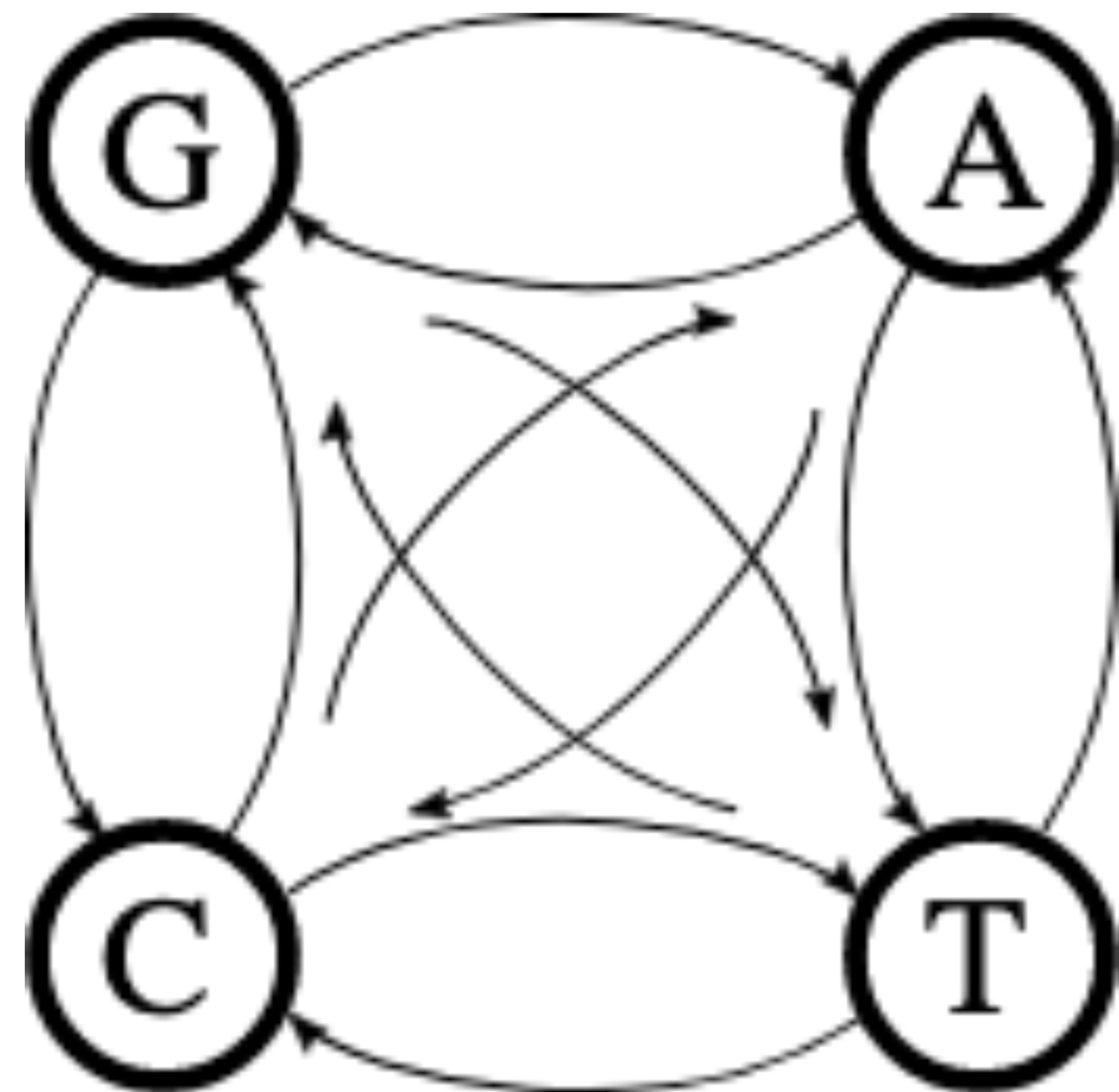
- یک نقطه ابتدایی x_0
- هر دفعه
- یک همسایه از توزیع $g(x' | x_t)$ انتخاب کن
- با احتمال تغییر را بپذیر

$$A(x', x_t) = \min \left(1, \frac{P(x')}{P(x_t)} \frac{g(x_t | x')}{g(x' | x_t)} \right)$$

$L(x') / L(x_t)$

1

DNA مدل تکامل



مدل تکامل مارکوف

احتمال مشاهده حالت‌ها در زمان t :

$$p'(t) = p(t)Q$$

جمع سطر = \circ

$$p(t) = p(0)e^{tQ}$$

توزیع پایایی یکتا

$$\pi e^{tQ} = \pi$$

بازگشت‌پذیر

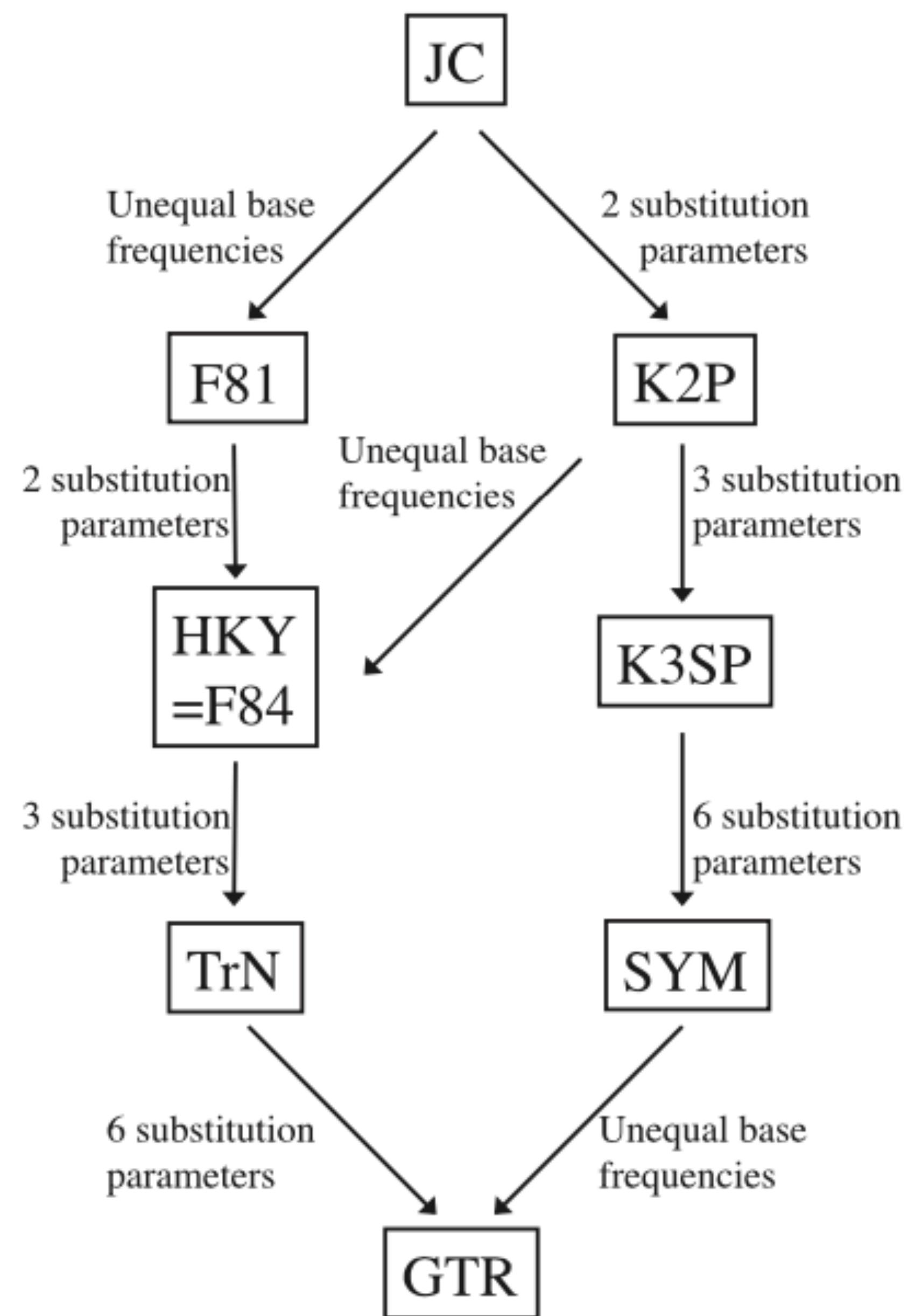
$$xe^{tQ} = y \Leftrightarrow ye^{tQ} = x$$

مدل JC69 (Jukes and Cantor) JC69

● یک پارامتر μ : نرخ جهش در یک واحد زمانی

$$P = \begin{pmatrix} \frac{1}{4} + \frac{3}{4}e^{-t\mu} & \frac{1}{4} - \frac{1}{4}e^{-t\mu} & \frac{1}{4} - \frac{1}{4}e^{-t\mu} & \frac{1}{4} - \frac{1}{4}e^{-t\mu} \\ \frac{1}{4} - \frac{1}{4}e^{-t\mu} & \frac{1}{4} + \frac{3}{4}e^{-t\mu} & \frac{1}{4} - \frac{1}{4}e^{-t\mu} & \frac{1}{4} - \frac{1}{4}e^{-t\mu} \\ \frac{1}{4} - \frac{1}{4}e^{-t\mu} & \frac{1}{4} - \frac{1}{4}e^{-t\mu} & \frac{1}{4} + \frac{3}{4}e^{-t\mu} & \frac{1}{4} - \frac{1}{4}e^{-t\mu} \\ \frac{1}{4} - \frac{1}{4}e^{-t\mu} & \frac{1}{4} - \frac{1}{4}e^{-t\mu} & \frac{1}{4} - \frac{1}{4}e^{-t\mu} & \frac{1}{4} + \frac{3}{4}e^{-t\mu} \end{pmatrix}$$

$$Q = \begin{pmatrix} * & \frac{\mu}{4} & \frac{\mu}{4} & \frac{\mu}{4} \\ \frac{\mu}{4} & * & \frac{\mu}{4} & \frac{\mu}{4} \\ \frac{\mu}{4} & \frac{\mu}{4} & * & \frac{\mu}{4} \\ \frac{\mu}{4} & \frac{\mu}{4} & \frac{\mu}{4} & * \end{pmatrix}$$



بسم الله الرحمن الرحيم

ڙنو ميڪ محاسباتي

جلسه ٥: بازسازی درخت تبارزایی (۲)

ترم پايز ١٤٠١-١٤٠٠

