

ВВЕДЕНИЕ В БИОИНФОРМАТИКУ

Лекция №7

Методы кластеризации.
Филогенетические деревья.
Множественные выравнивания.

Новоселецкий Валерий Николаевич
к.ф.-м.н., доц. каф. биоинженерии
valery.novoseletsky@yandex.ru

Сайт курса <http://intbio.org/bioinf2020-21>

Множественное выравнивание последовательностей

Цели:

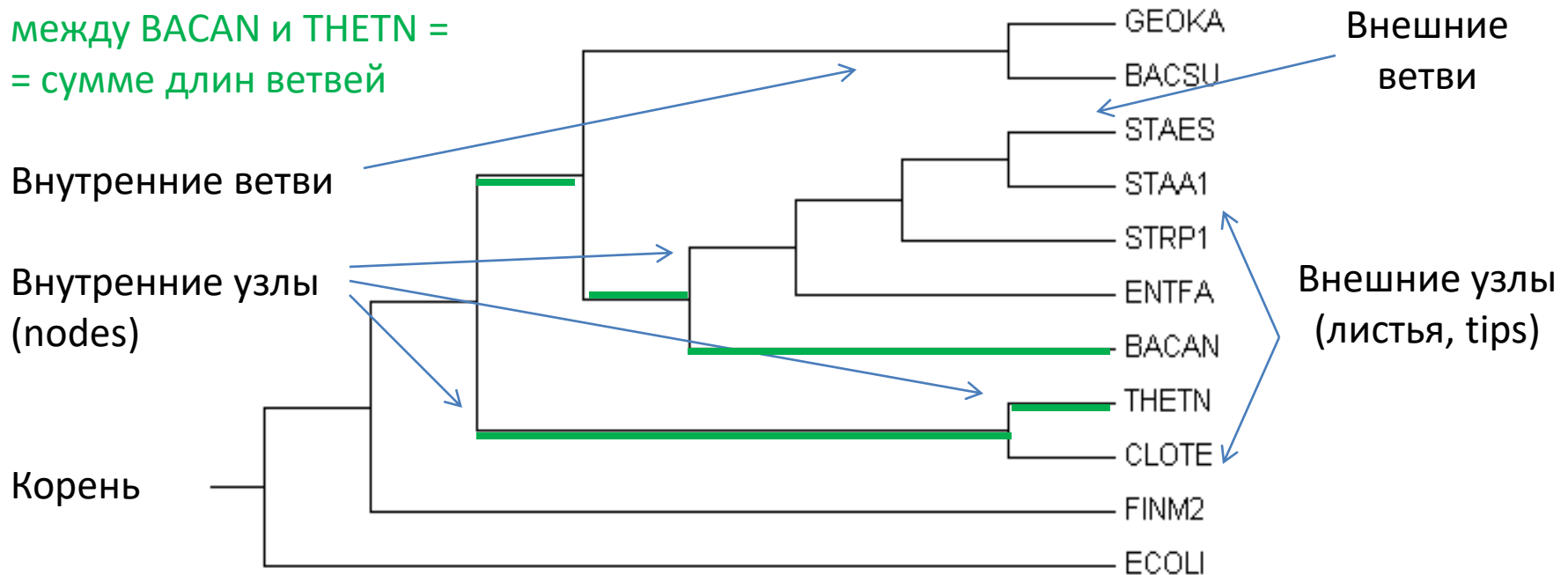
- Построение филогенетических деревьев
- **Выявление консервативных остатков и мотивов**
- **Построение профилей (визуализация)**
- **Итеративное выявление удаленной гомологии**
- ...

Алгоритмы:

- **Динамическое программирование – не годится**
- Прогрессивное выравнивание
- Скрытые марковские модели
- Квантовые компьютеры?

Филогенетические деревья: определения

Эволюционное расстояние
между BACAN и THETN =
= сумме длин ветвей



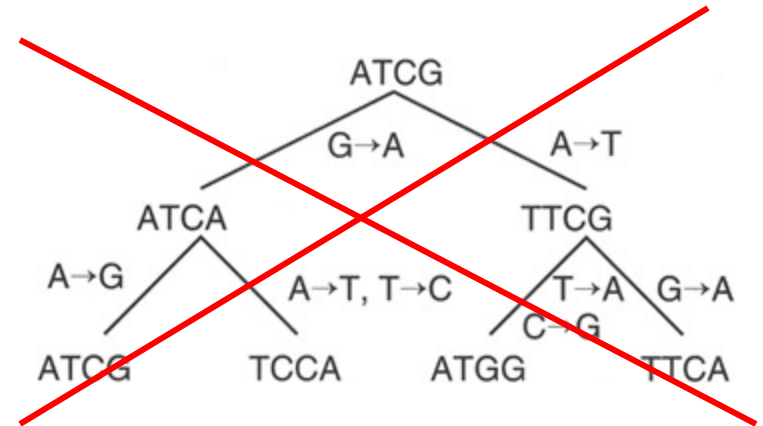
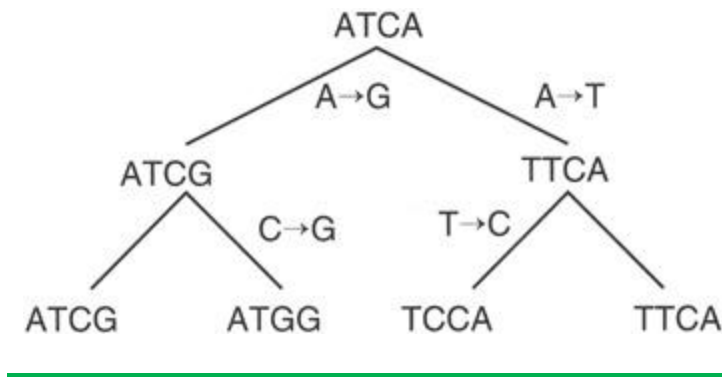
Кладограмма — филогенетическое дерево, не содержащее информации о длинах ветвей.

Филограмма — филогенетическое дерево, содержащее информацию о длинах ветвей; эти длины представляют изменение некой характеристики.

Филогенетические деревья: методы оценки

Максимальная экономия (Fitch, 1971) – критерий оптимальности, согласно которому **предпочтительнее деревья с меньшим суммарным числом мутаций**.

Однако алгоритма быстрого построения такого дерева не существует.



Метод максимального правдоподобия учитывает не просто число мутаций, но и их вероятность.

Деревья: свойства

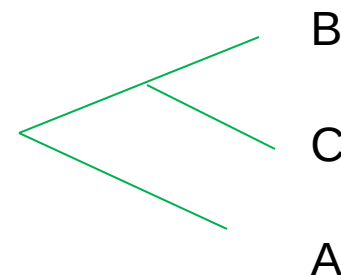
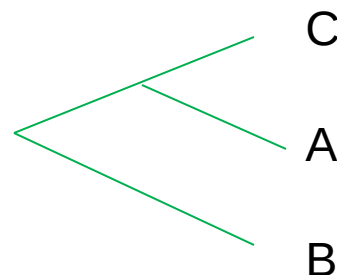
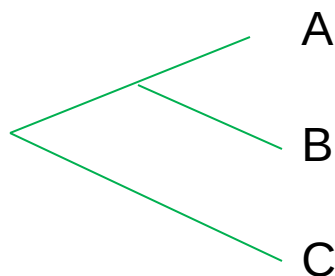
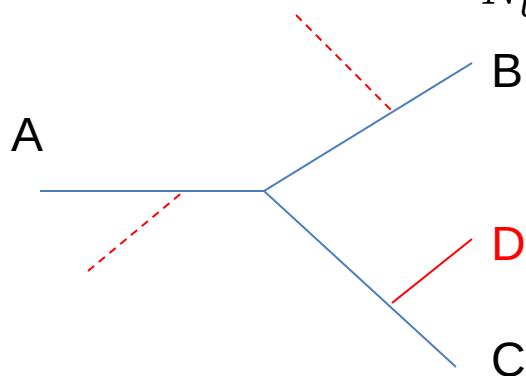
Число деревьев

неукорененных

$$N_U = (2n - 5)!!$$

укорененных

$$N_R = (2n - 3)!!$$



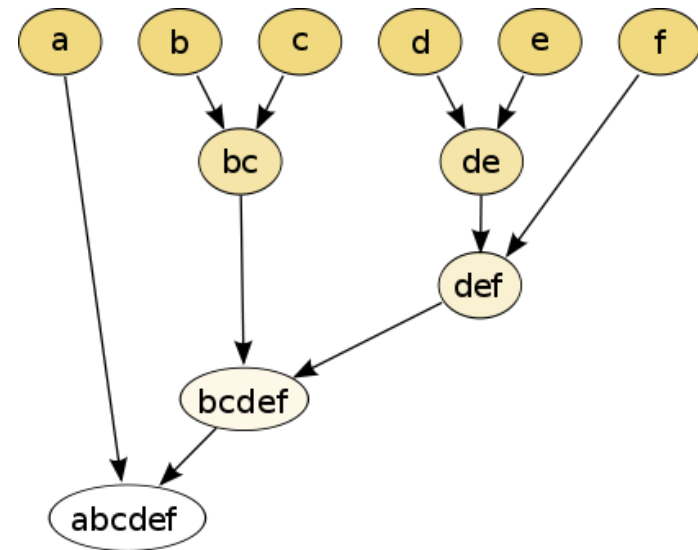
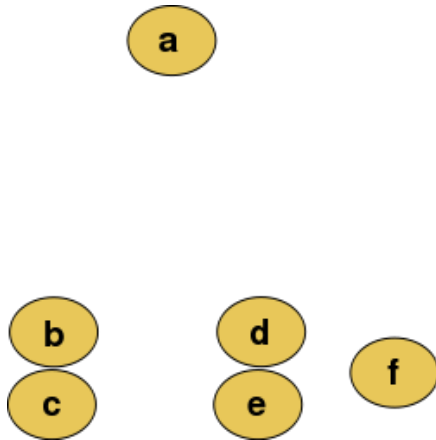
Число листьев n	Число неукорененных деревьев N_U	Число укорененных деревьев N_R
3	1	3
4	3	15
10	2027025	34459425
20	2,21643E+20	8,20079E+21
	Возраст Земли	1,4E+17 секунд

С точки зрения филогении, правильное только одно! Как его отыскать?

Методы иерархической кластеризации. UPGMA

Unweighted Pair Group Method with Arithmetic mean (Sokal, Michener, 1958)

- метод невзвешенной группировки с арифметическим средним
- пример алгоритма иерархической кластеризации



Расстояние между элементами

$$\|a - b\|_2 = \sqrt{\sum_i (a_i - b_i)^2}$$

Расстояние между кластерами

$$\frac{1}{|\mathcal{A}| \cdot |\mathcal{B}|} \sum_{x \in \mathcal{A}} \sum_{y \in \mathcal{B}} d(x, y).$$

Методы иерархической кластеризации. UPGMA

Дан набор объектов S_k , где для каждой пары (S_i, S_j) установлена мера сходства $L(S_i, S_j)$. Для построения дерева выбирают два наиболее близких объекта (S_m, S_n) и добавляют вершину, изображающую их общего «предка» (S_{mn}) . Затем замещают эти два объекта группой, содержащий обоих, и присваивают расстояниям от этой пары до остальных объектов S_k средние значения от каждого из элементов этой группы до S_k :

$$L(S_{mn}, S_k) = \frac{L(S_m, S_k) + L(S_n, S_k)}{2}$$

В случае объединения кластеров C_i и C_j с образованием кластера C_k , содержащего $n_i + n_j = n_k$ элементов, расстояние от кластера C_k до остальных кластеров C_m вычисляется как

$$L(C_k, C_m) = \frac{n_i L(C_i, C_m) + n_j L(C_j, C_m)}{n_i + n_j}$$

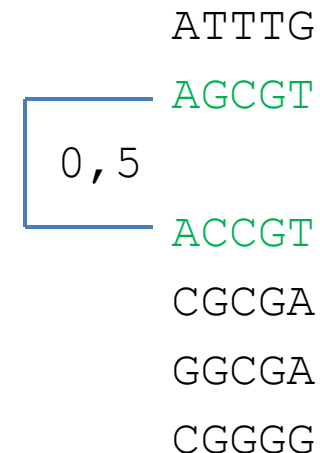
Методы иерархической кластеризации. UPGMA

Дано 6 последовательностей – ATTTG, AGCGT, ACCGT, CGCGA, GGCGA, CGGGG.

Используя **расстояние по Хэммингу**, получаем матрицу расстояний D_0 :

D0	ATTTG	AGCGT	ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	4	4	5	5	4
AGCGT		0	1	2	2	3
ACCGT			0	3	3	4
CGCGA				0	1	2
GGCGA					0	3
CGGGG						0

Объединяем первую пару элементов



Методы иерархической кластеризации. UPGMA

Дано 6 последовательностей – ATTTG, AGCGT, ACCGT, CGCGA, GGCGA, CGGGG.

D0	ATTTG	AGCGT	ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	4	4	5	5	4
AGCGT		0	1	2	2	3
ACCGT			0	3	3	4
CGCGA				0	1	2
GGCGA					0	3
CGGGG						0

Матрицу D0 преобразуем в матрицу D1:

D1	ATTTG	AGCGT, ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	$(4+4)/2=4$	5	5	4
AGCGT, ACCGT		0	$(2+3)/2=2,5$	$(2+3)/2=2,5$	$(3+4)/2=3,5$
CGCGA			0	1	2
GGCGA				0	3
CGGGG					0

Методы иерархической кластеризации. UPGMA

D1	ATTTG	AGCGT, ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	$(4+4)/2=4$	5	5	4
AGCGT, ACCGT		0	$(2+3)/2=2,5$	$(2+3)/2=2,5$	$(3+4)/2=3,5$
CGCGA			0	1	2
GGCGA				0	3
CGGGG					0

Объединяем вторую пару элементов



Методы иерархической кластеризации. UPGMA

D1	ATTTG	AGCGT, ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	$(4+4)/2=4$	5	5	4
AGCGT, ACCGT		0	$(2+3)/2=2,5$	$(2+3)/2=2,5$	$(3+4)/2=3,5$
CGCGA			0	1	2
GGCGA				0	3
CGGGG					0

Матрицу D1 преобразуем в матрицу D2:

D2	ATTTG	AGCGT, ACCGT	CGCGA, GGCGA	CGGGG
ATTTG	0	4	$(5+5)/2=5$	4
AGCGT, ACCGT		0	$(2,5+2,5)/2=2,5$	3,5
CGCGA, GGCGA			0	$(2+3)/2=2,5$
CGGGG				0

Методы иерархической кластеризации. UPGMA

D2	ATTTG	AGCGT, ACCGT	CGCGA, GGCGA	CGGGG
ATTTG	0	4	$(5 + 5)/2 = 5$	4
AGCGT, ACCGT		0	$(2,5 + 2,5)/2 = 2,5$	3,5
CGCGA, GGCGA			0	$(2 + 3)/2 = 2,5$
CGGGG				0

Добавляем третий элемент
во второй кластер



Методы иерархической кластеризации. UPGMA

D2	ATTTG	AGCGT, ACCGT	CGCGA, GGCGA	CGGGG
ATTTG	0	4	$(5 + 5)/2 = 5$	4
AGCGT, ACCGT		0	$(2,5 + 2,5)/2 = 2,5$	3,5
CGCGA, GGCGA			0	$(2 + 3)/2 = 2,5$
CGGGG				0

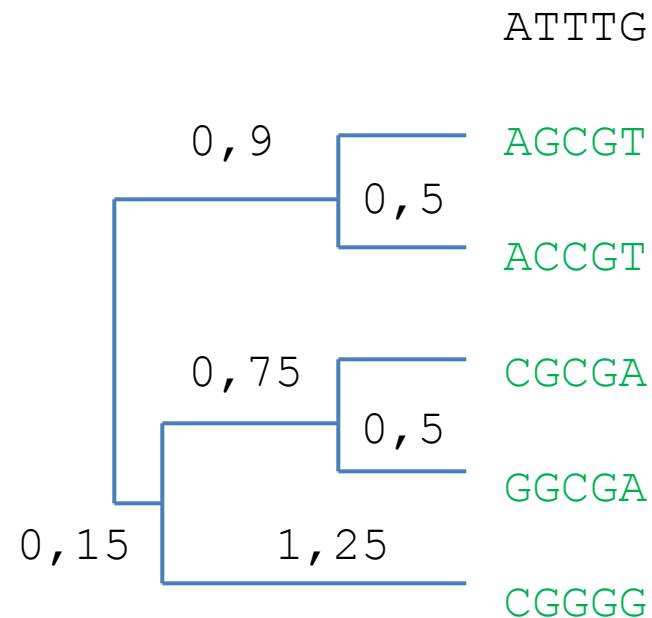
Матрицу D2 преобразуем в матрицу D3:

D3	ATTTG	AGCGT, ACCGT	(CGCGA, GGCGA), CGGGG
ATTTG	0	4	$(2*5 + 4)/3 = 4,7$
AGCGT, ACCGT		0	$(2*2,5 + 3,5)/3 = 2,8$
(CGCGA, GGCGA), CGGGG			0

Методы иерархической кластеризации. UPGMA

D3	ATTTG	AGCGT, ACCGT	(CGCGA, GGCGA), CGGGG
ATTTG	0	4	$(2*5 + 4)/3 = 4,7$
AGCGT, ACCGT		0	$(2*2,5 + 3,5)/3 = 2,8$
(CGCGA, GGCGA), CGGGG			0

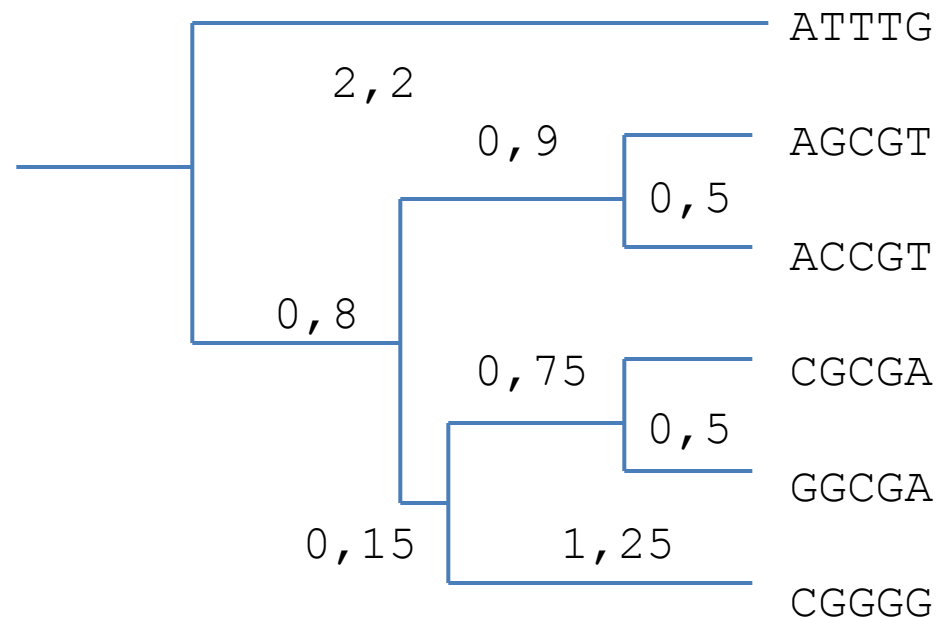
Объединяем кластеры



Методы иерархической кластеризации. UPGMA

Рассчитываем последнюю матрицу D4:

D4	ATTTG	((CGCGA, GGCGA), CGGGG), (AGCGT, ACCGT)
ATTTG	0	$(4*2 + 4,7*3)/5 = 4,4$
((CGCGA, GGCGA), CGGGG), (AGCGT, ACCGT)		0



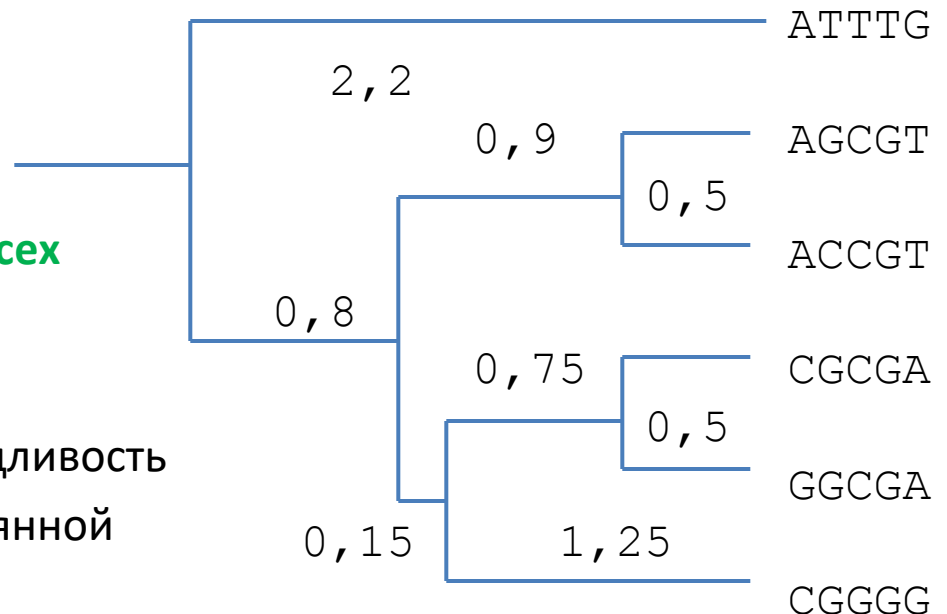
И добавляем последний элемент

Методы иерархической кластеризации. UPGMA

D4	ATTTG	((CGCGA, GGCGA), CGGGG), (AGCGT, ACCGT)
ATTTG	0	$(4*2 + 4,7*3)/5 = 4,4$
((CGCGA, GGCGA), CGGGG), (AGCGT, ACCGT)		0

Длины ветвей установлены так, что
**расстояние от корня одинаково для всех
 листьев - ультраметричность.**

Метод UPGMA подразумевает справедливость
гипотезы молекулярных часов (постоянной
 скорости эволюции).



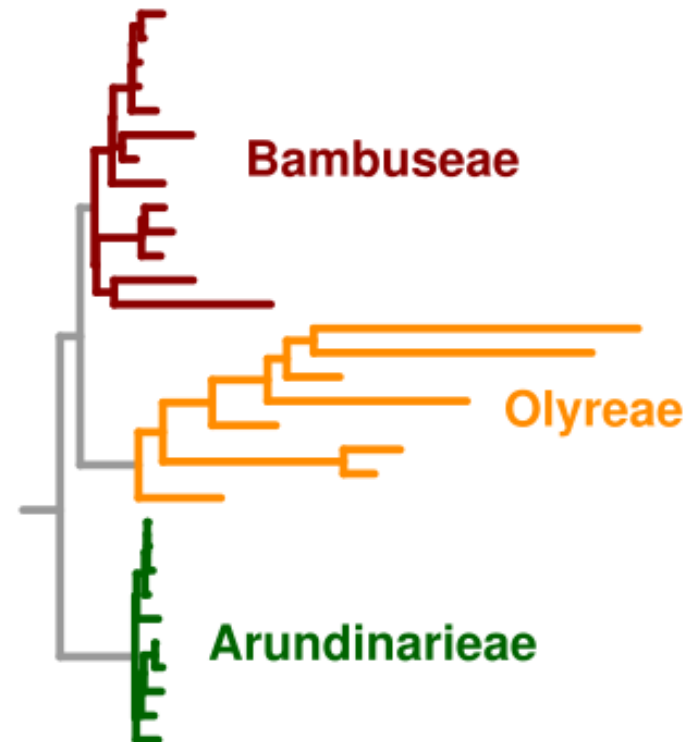
Гипотеза молекулярных часов

Сопоставления числа различий в аминокислотных последовательностях **гемоглобинов** млекопитающих и срока дивергенции (Э. Цукеркандль, Л. Полинг, 1962).

Сопоставления идентичности последовательностей **цитохрома с** у рыб, птиц и млекопитающих (Э. Марголиаш, 1963).

Гипотеза широко распространена, хотя и имеется довольно много примеров, ей противоречащих.

Трудности калибровки



Методы иерархической кластеризации.

Метод связи между соседями

Соседи – последовательности, расположенные в дереве через один узел (A и B, C и D).

Считая длины ветвей известными, видим, что:

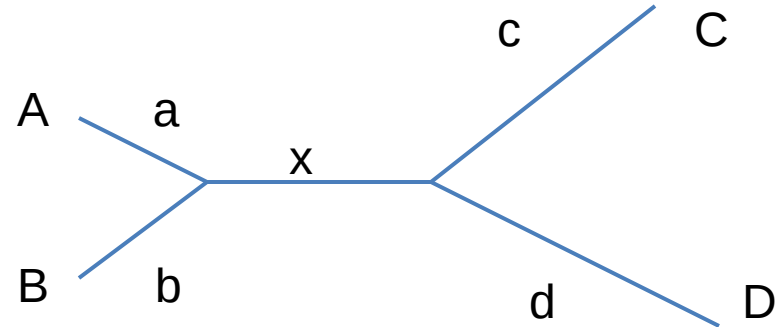
$$L_{AC} + L_{BD} = L_{AD} + L_{BC} = a + b + c + d + 2x = L_{AB} + L_{CD} + 2x$$

Очевидно, что

$$L_{AB} + L_{CD} < L_{AC} + L_{BD}$$

$$L_{AB} + L_{CD} < L_{AD} + L_{BC}$$

Условие четырех точек (1974)

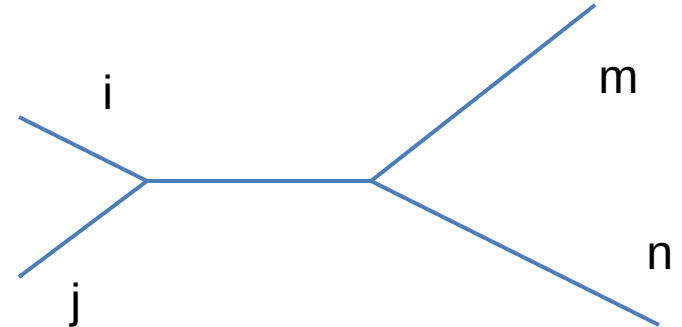


Если известны расстояния между последовательностями, но неизвестны эволюционные отношения, то **метод позволяет установить топологию филогенетического дерева**, т.е. как раз отношения.

Методы иерархической кластеризации.

Метод связи между соседями

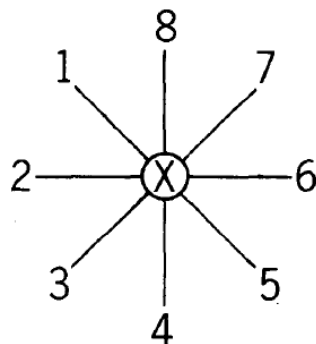
На большее число последовательностей обобщается путем рассмотрения всех четверок и определением тех из них, для которых суммы расстояний минимальны.



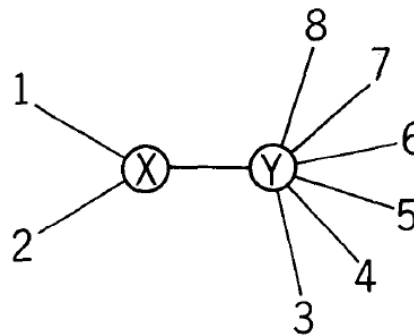
Если $\min(L_{ij} + L_{mn}, L_{im} + L_{jn}, L_{in} + L_{jm}) = L_{ij} + L_{mn}$, то

N	...	<i>i</i>	<i>j</i>	...	<i>m</i>	<i>n</i>
...						
<i>i</i>			+1		+0	+0
<i>j</i>		+1			+0	+0
...						
<i>m</i>		+0	+0			+1
<i>n</i>		+0	+0		+1	

Методы иерархической кластеризации. NJ



(a)



(b)

Neighbor joining – метод присоединения соседей (Saitou, Nei, 1987)

- еще один алгоритм иерархической кластеризации.

Цель: построить **дерево с минимальной суммой длин ветвей**.

Пусть

D_{ij} – расстояние между таксонами i и j ,

L_{ab} – длина ветви между узлами a и b

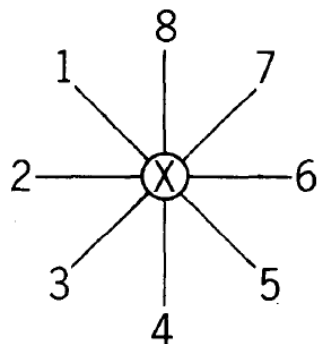
D – известная матрица расстояний

L нужно найти

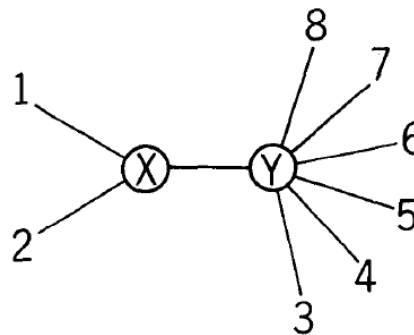
Тогда для суммы длин ветвей дерева на рис. (a) выполняется

$$S_0 = \sum_{i=1}^N L_{iX} = \frac{1}{N-1} \sum_{i < j} D_{ij} \quad (\text{поскольку } D_{ij} = L_{iX} + L_{Xj}) \quad (1)$$

Методы иерархической кластеризации. NJ



(a)



(b)

Выделив пару таксонов 1 и 2 и добавив узел Y, для суммы расстояний от таксонов 1 и 2 до всех остальных имеем (рис. (b)):

$$\sum_{k=3}^N (D_{1k} + D_{2k}) = (N-2)L_{1X} + (N-2)L_{2X} + 2(N-2)L_{XY} + 2\sum_{i=3}^N L_{iY} \quad (2)$$

А для суммы всех ветвей дерева (рис. (b))

$$S_{12} = L_{XY} + L_{1X} + L_{2X} + \sum_{i=3}^N L_{iY} \quad (3)$$

Теперь надо так выбрать таксоны 1 и 2, чтобы сумма всех ветвей дерева была минимальна

Методы иерархической кластеризации. NJ

Итак, имеем

$$\sum_{k=3}^N (D_{1k} + D_{2k}) = (N-2)L_{1X} + (N-2)L_{2X} + 2(N-2)L_{XY} + 2\sum_{i=3}^N L_{iY} \quad (2)$$

$$S_{12} = L_{XY} + L_{1X} + L_{2X} + \sum_{i=3}^N L_{iY} \quad (3)$$

Выражаем L_{XY} из (2) и подставляем в (3)

$$L_{XY} = \frac{1}{2(N-2)} \left(\sum_{k=3}^N (D_{1k} + D_{2k}) - (N-2)(L_{1X} + L_{2X}) - 2\sum_{i=3}^N L_{iY} \right)$$

$$S_{12} = \frac{1}{2(N-2)} \left(\sum_{k=3}^N (D_{1k} + D_{2k}) - (N-2)(L_{1X} + L_{2X}) - 2\sum_{i=3}^N L_{iY} \right) +$$

$$+ (L_{1X} + L_{2X}) + \sum_{i=3}^N L_{iY} = \frac{\sum_{k=3}^N (D_{1k} + D_{2k})}{2(N-2)} + \frac{L_{1X} + L_{2X}}{2} + \frac{N-3}{N-2} \sum_{i=3}^N L_{iY}$$

Методы иерархической кластеризации. NJ

Заметим, что $L_{1X} + L_{2X} = D_{12}$, а также, по аналогии с (1),

$$\sum_{i=3}^N L_{iY} = \frac{1}{N-3} \sum_{3 \leq i < j} D_{ij}$$

Теперь S_{12} принимает вид

$$S_{12} = \frac{\sum_{i=3}^N (D_{1i} + D_{2i})}{2(N-2)} + \frac{D_{12}}{2} + \frac{\sum_{3 \leq i < j} D_{ij}}{N-2}$$

Для произвольных соседей k и l получим

$$S_{kl} = \frac{\sum_{i \neq k, l}^N (D_{ki} + D_{li})}{2(N-2)} + \frac{D_{kl}}{2} + \frac{\sum_{i < j, i \neq k, l} D_{ij}}{N-2}$$

Здесь все слагаемые так или иначе зависят от k и l , что неудобно.

Попробуем упростить.

Методы иерархической кластеризации. NJ

$$\begin{aligned}
 S_{kl} &= \frac{\sum_{i \neq k, l}^N (D_{ki} + D_{li})}{2(N-2)} + \frac{D_{kl}}{2} + \frac{\sum_{i < j, i \neq k, l} D_{ij}}{N-2} = \\
 &= \frac{\sum_i^N D_{ki} - D_{kk} + \sum_i^N D_{li} - D_{ll} + 2 \left(\sum_{i < j} D_{ij} - \sum_i^N D_{ki} - \sum_i^N D_{li} \right)}{2(N-2)} + \frac{D_{kl}}{2} = \\
 &= \frac{2 \sum_{i < j} D_{ij} - \sum_i^N D_{ki} - \sum_i^N D_{li}}{2(N-2)} + \frac{D_{kl}}{2}
 \end{aligned}$$

Заметим, что $\sum_{i < j} D_{ij}$ — не зависит от k и l .

Поэтому домножив на $2(N-2)$ и вычтя эту постоянную,

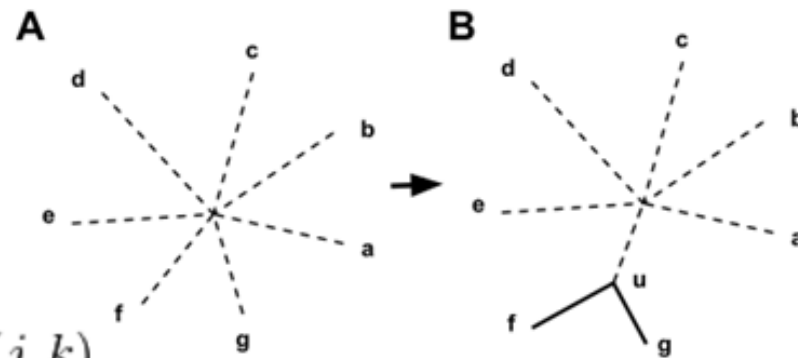
получим $Q_{kl} = D_{kl}(N-2) - \sum_i^N D_{ki} - \sum_i^N D_{li}$

Методы иерархической кластеризации. NJ

Пошаговая реализация:

1. По текущей матрице расстояний $d(i, j)$ рассчитывается Q -матрица

$$Q(i, j) = (n - 2)d(i, j) - \sum_{k=1}^n d(i, k) - \sum_{k=1}^n d(j, k)$$



2. Ищется пара различных таксонов i и j , для которых значение $Q(i, j)$ наименьшее. Эти таксоны присоединяются к новому узлу u , который, в свою очередь, соединяется с центральным <виртуальным> узлом.

3. Рассчитывается расстояние от каждого из присоединенных таксонов к новому узлу

$$\delta(f, u) = \frac{1}{2}d(f, g) + \frac{1}{2(n - 2)} \left[\sum_{k=1}^n d(f, k) - \sum_{k=1}^n d(g, k) \right]$$

Заметим, что при таком определении

$$\delta(f, u) + \delta(g, u) = d(f, g)$$

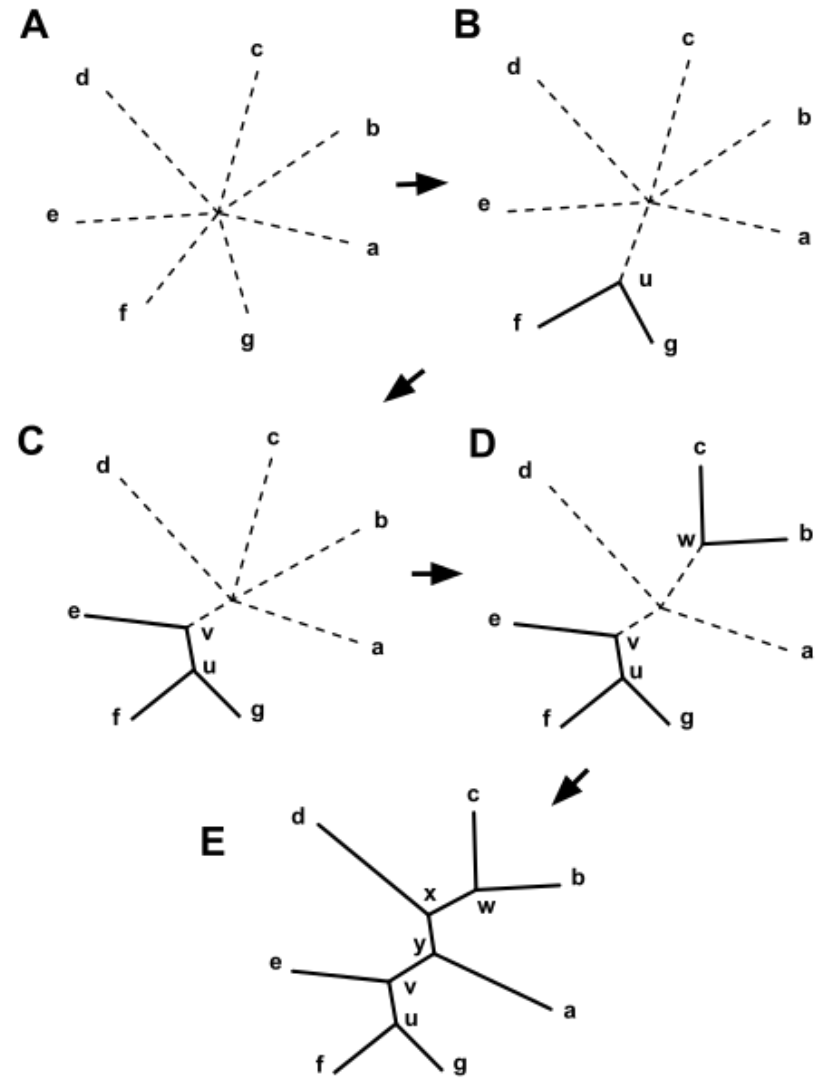
Методы иерархической кластеризации. NJ

4. Рассчитывается расстояние от каждого из оставшихся таксонов до нового узла

$$d(u, k) = \frac{1}{2}[d(f, k) + d(g, k) - d(f, g)]$$

5. Алгоритм запускается снова, заменяя пару соединенных таксонов на новый узел и используя расстояния, посчитанные на предыдущих шагах.

Длины ветвей, выходящих из одного узла в общем случае неравны.



Методы иерархической кластеризации. NJ

Те же 6 последовательностей, та же матрица расстояний:

D0	ATTTG	AGCGT	ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	4	4	5	5	4
AGCGT		0	1	2	2	3
ACCGT			0	3	3	4
CGCGA				0	1	2
GGCGA					0	3
CGGGG						0

$$Q(i, j) = (n - 2)d(i, j) - \sum_{k=1}^n d(i, k) - \sum_{k=1}^n d(j, k) \quad n = 6$$

Q0	ATTTG	AGCGT	ACCGT	CGCGA	GGCGA	CGGGG
ATTTG		-18	-21	-15	-16	-22
AGCGT			-23	-17	-18	-16
ACCGT				-16	-17	-15
CGCGA					-23	-21
GGCGA						-18
CGGGG						

Методы иерархической кластеризации. NJ

D0	ATTTG	AGCGT	ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	4	4	5	5	4
AGCGT		0	1	2	2	3
ACCGT			0	3	3	4
CGCGA				0	1	2
GGCGA					0	3
CGGGG						0

$$\delta(f, u) = \frac{1}{2}d(f, g) + \frac{1}{2(n-2)} \left[\sum_{k=1}^n d(f, k) - \sum_{k=1}^n d(g, k) \right]$$

δ1	AGCGT, ACCGT
AGCGT	0, 125
ACCGT	0, 875

Методы иерархической кластеризации. NJ

D0	ATTTG	AGCGT	ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	4	4	5	5	4
AGCGT		0	1	2	2	3
ACCGT			0	3	3	4
CGCGA				0	1	2
GGCGA					0	3
CGGGG						0

$$d(u, k) = \frac{1}{2}[d(f, k) + d(g, k) - d(f, g)]$$

D1	ATTTG	AGCGT, ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	3,5	5	5	4
AGCGT, ACCGT		0	2	2	3
CGCGA			0	1	2
GGCGA				0	3
CGGGG					0

Методы иерархической кластеризации. NJ

D1	ATTTG	AGCGT, ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	3,5	5	5	4
AGCGT, ACCGT		0	2	2	3
CGCGA			0	1	2
GGCGA				0	3
CGGGG					0

$$Q(i, j) = (n - 2)d(i, j) - \sum_{k=1}^n d(i, k) - \sum_{k=1}^n d(j, k) \quad n = 5$$

Q1	ATTTG	AGCGT, ACCGT	CGCGA	GGCGA	CGGGG
ATTTG		-17,5	-12,5	-13,5	-17,5
AGCGT, ACCGT			-14,5	-15,5	-13,5
CGCGA				-18	-16
GGCGA					-14
CGGGG					

Методы иерархической кластеризации. NJ

D1	ATTTG	AGCGT, ACCGT	CGCGA	GGCGA	CGGGG
ATTTG	0	3,5	5	5	4
AGCGT, ACCGT		0	2	2	3
CGCGA			0	1	2
GGCGA				0	3
CGGGG					0

$\delta 2$	CGCGA, GGCGA
CGCGA	0,333
GGCGA	0,667

$$\delta(f, u) = \frac{1}{2}d(f, g) + \frac{1}{2(n-2)} \left[\sum_{k=1}^n d(f, k) - \sum_{k=1}^n d(g, k) \right]$$

$$d(u, k) = \frac{1}{2}[d(f, k) + d(g, k) - d(f, g)]$$

D2	ATTTG	AGCGT, ACCGT	CGCGA, GGCGA	CGGGG
ATTTG	0	3,5	4,5	4
AGCGT, ACCGT		0	1,5	3
CGCGA, GGCGA			0	2
CGGGG				0

Методы иерархической кластеризации. NJ

D2	ATTTG	AGCGT, ACCGT	CGCGA, GGCGA	CGGGG
ATTTG	0	3,5	4,5	4
AGCGT, ACCGT		0	1,5	3
CGCGA, GGCGA			0	2
CGGGG				0

Q2	ATTTG	AGCGT, ACCGT	CGCGA, GGCGA	CGGGG
ATTTG		-13	-11	-13
AGCGT, ACCGT			-13	-11
CGCGA, GGCGA				-13
CGGGG				

δ3	(CGCGA, GGCGA), CGGGG
CGCGA, GGCGA	0,75
CGGGG	1,25

Методы иерархической кластеризации. NJ

D2	ATTTG	AGCGT, ACCGT	CGCGA, GGC GA	CGGGG
ATTTG	0	3,5	4,5	4
AGCGT, ACCGT		0	1,5	3
CGCGA, GGC GA			0	2
CGGGG				0

Q2	ATTTG	AGCGT, ACCGT	CGCGA, GGC GA	CGGGG
ATTTG		-13	-11	-13
AGCGT, ACCGT			-13	-11
CGCGA, GGC GA				-13
CGGGG				

D3	ATTTG	AGCGT, ACCGT	(CGCGA, GGC GA), CGGGG
ATTTG	0	3,5	3,25
AGCGT, ACCGT		0	1,25
(CGCGA, GGC GA), CGGGG			0

Методы иерархической кластеризации. NJ

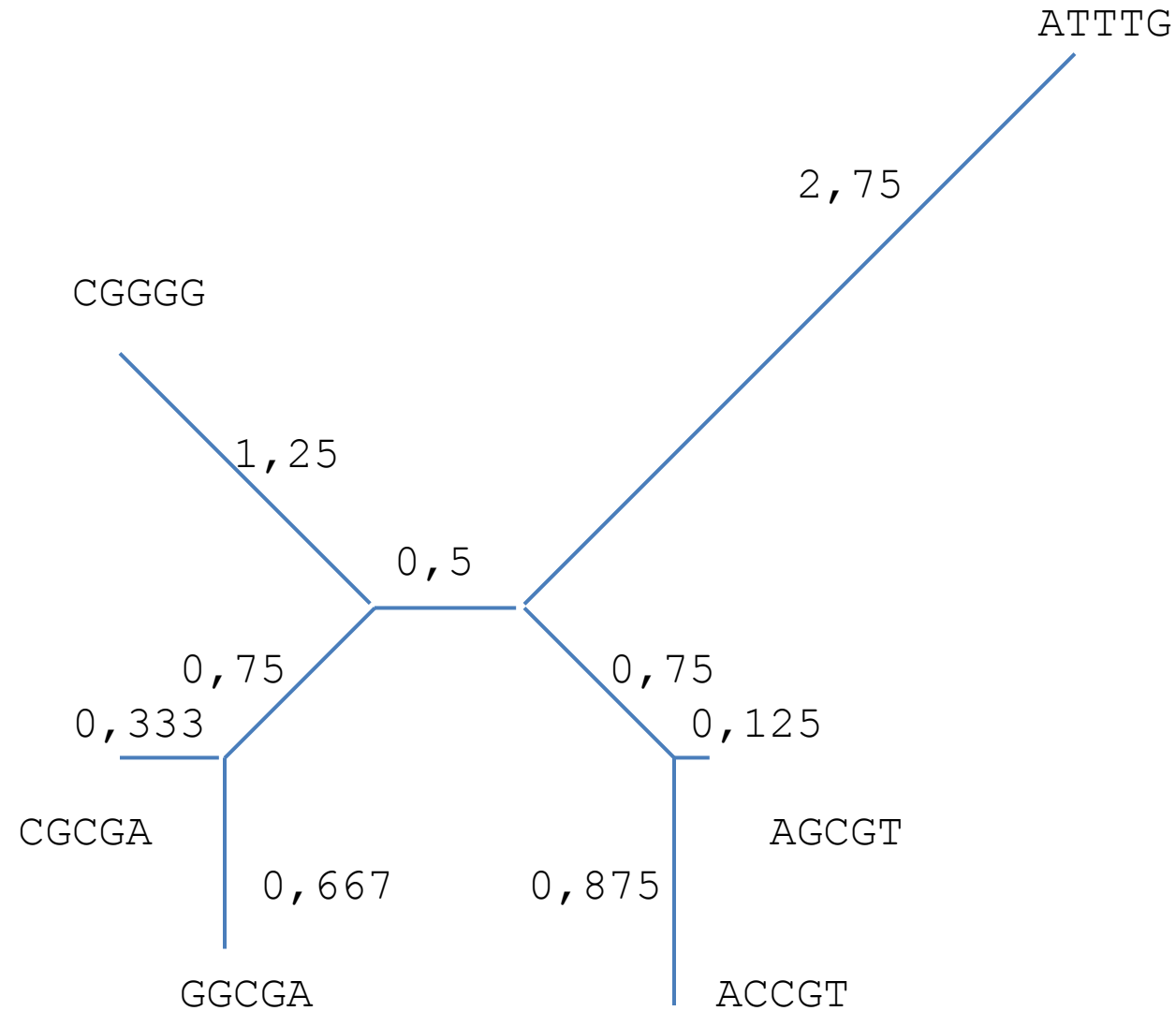
D3	ATTTG	AGCGT, ACCGT	(CGCGA, GGCGA), CGGGG
ATTTG	0	3,5	3,25
AGCGT, ACCGT		0	1,25
(CGCGA, GGCGA), CGGGG			0

Q3	ATTTG	AGCGT, ACCGT	(CGCGA, GGCGA), CGGGG
ATTTG		-8	-8
AGCGT, ACCGT			-8
(CGCGA, GGCGA), CGGGG			

δ4	ATTTG, AGCGT, ACCGT
ATTTG	0,75
AGCGT, ACCGT	1,25

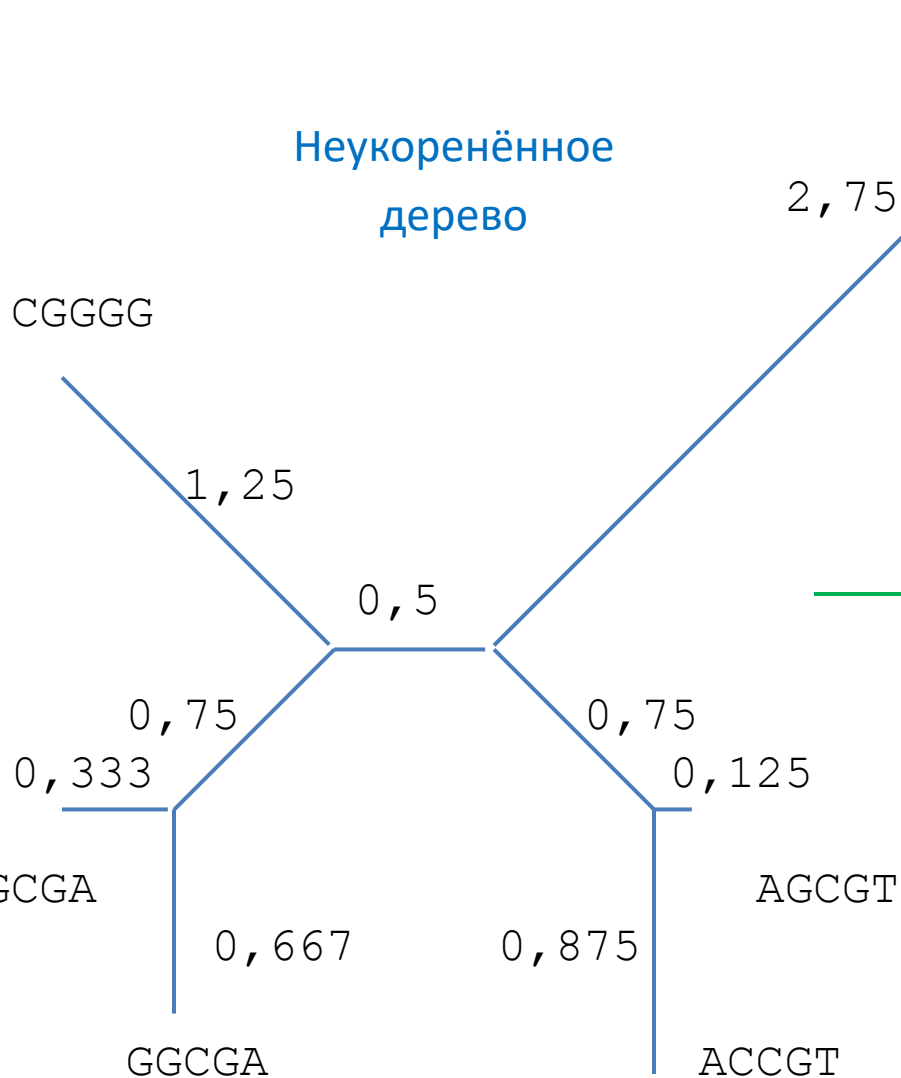
D4	AGCGT, ACCGT, ATTTG	CGCGA, GGCGA, CGGGG
AGCGT, ACCGT, ATTTG	0	0,5
CGCGA, GGCGA, CGGGG	0,5	0

Методы иерархической кластеризации. NJ

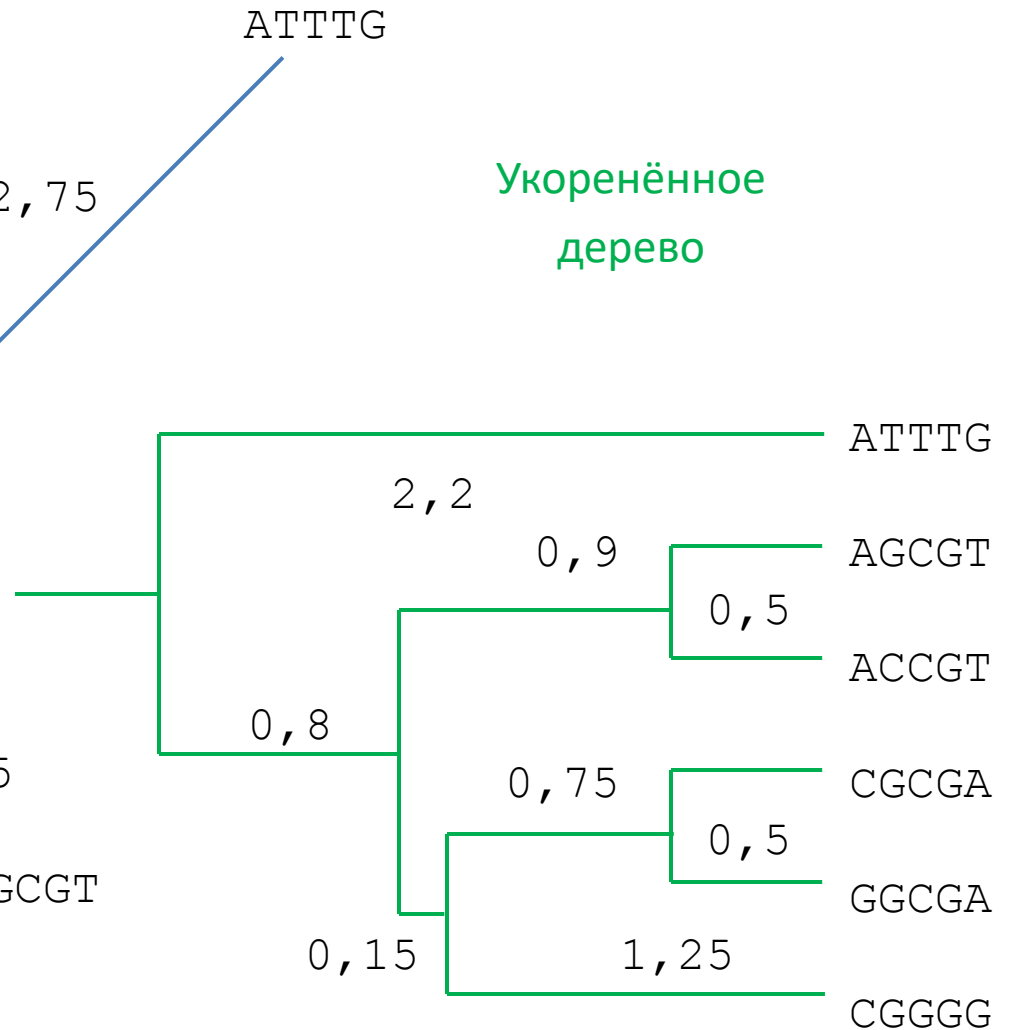


Методы иерархической кластеризации. NJ и UPGMA

Неукоренённое
дерево



Укоренённое
дерево



Множественное выравнивание последовательностей

Цели:

- Построение филогенетических деревьев
- Выявление консервативных остатков и мотивов
- Построение профилей (визуализация)
- Итеративное выявление удаленной гомологии
- ...

Алгоритмы:

- Динамическое программирование – не годится
- Прогрессивное выравнивание
- Скрытые марковские модели
- Квантовые компьютеры?

Прогрессивное выравнивание. Clustal

Clustal (Higgins and Sharp, 1988) выполняет постепенное выравнивание все новых последовательностей, начиная с наиболее <эволюционно> близких, ориентируясь на предварительно построенное на основании парных выравниваний (**сравнений**) филогенетическое дерево.

Алгоритм:

- 1) Экспресс-оценка сходства двух последовательностей вычисляется как **число совпадающих остатков в словах длины K** ($K = 1-2$ для белковых последовательностей и $2-4$ для нуклеотидных) за вычетом штрафа за сделанные вставки.
- 2) Методом **UPGMA** (позже **NJ**) рассчитывается **направляющее дерево**, по которому затем рассчитываются веса последовательностей, причем более близкие последовательности получают меньшие веса.
- 3) Согласно **направляющего дерева** выбираются наиболее близкие последовательности и выполняется их выравнивание методом динамического программирования с использованием матрицы замен и штрафов за открытие/расширение вставок, с полученным выравниванием сопоставляются все новые последовательности.

Прогрессивное выравнивание. Clustal

Особое внимание уделено значениям штрафов за вставки. Введена их зависимость от:

- А) типа сопоставляемого и предшествующего остатков;
- Б) степени близости последовательностей;
- В) длин рассматриваемых последовательностей;
- Г) наличия вставок в уже имеющемся выравнивании;
- Д) характера аминокислотной последовательности.

One gap, always gap

Текущие версии: **ClustalW2** и **Clustal Omega** (использует СММ)



Clustal: Multiple Sequence Alignment
Multiple alignment of nucleic acid and protein sequences





Clustal Omega

- Latest version of Clustal - fast and scalable (can align hundreds of thousands of sequences in hours), greater accuracy due to new HMM alignment engine
- Command line/web server only (GUI public beta available soon)



ClustalW/ClustalX

- "Classic Clustal"
- GUI (ClustalX), command line (ClustalW), web server versions available

Итеративное выравнивание. MUSCLE

MUSCLE (MUltiple Sequence Comparison by Log-Expectation) (Edgar, 2004)

Три стадии:

- 1) **быстрое** «черновое» **множественное выравнивание** (попарные глобальные выравнивания; оценка сходства как доля совпадающих позиций; построение направляющего дерева методом **UPGMA** или **NJ**; прогрессивное выравнивание)
- 2) **улучшенное множественное выравнивание** (оценка сходства как доля совпадающих позиций в текущем множественном выравнивании; построение направляющего дерева через построение матрицы расстояний по Кимуре и ее кластеризацию; сравнение текущего дерева с построенным ранее; пересчет выравнивания для отличающихся узлов; повторение до сходимости)
- 3) **уточнение выравнивания** (удаление произвольного узла для разбиения дерева на два; построение профилей для каждого поддеревя и их выравнивание; расчет суммы парных оценок в получающемся множественном выравнивании; перебор всех узлов от листьев к корню и выбор выравнивания с максимальной суммой)

<http://www.drive5.com/muscle/>

35000+ цитирований (2020)

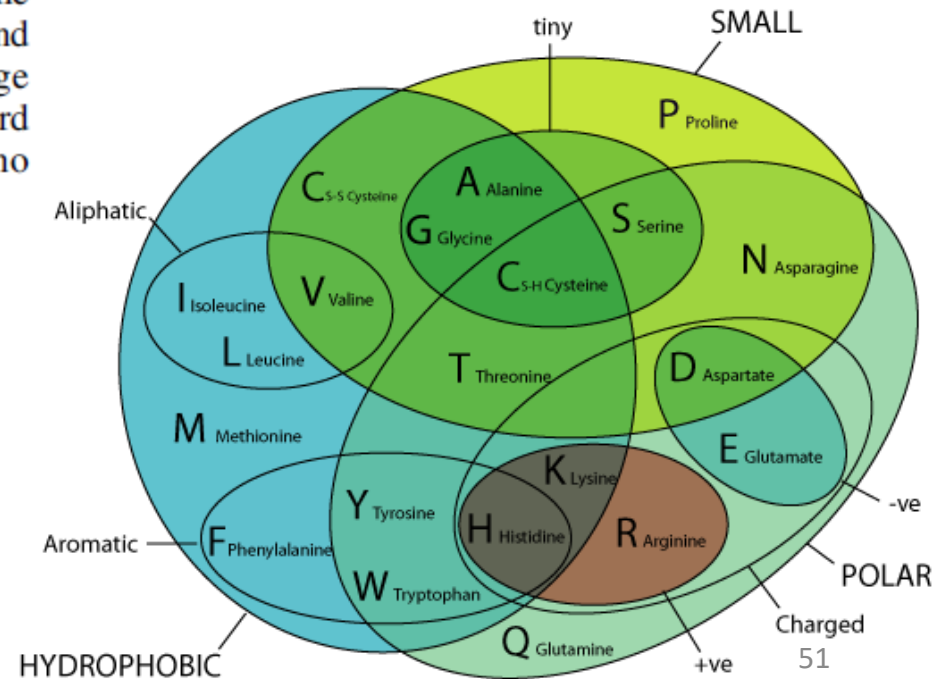
MAFFT (multiple alignment using fast Fourier transform)

METHODS

Group-to-group alignments by FFT

The frequency of amino acid substitutions strongly depends on the difference of physico-chemical properties, particularly volume and polarity, between the amino acid pair involved in the substitution (16). Substitutions between physico-chemically similar amino acids tend to preserve the structure of proteins, and such neutral substitutions have been accumulated in molecules during evolution (17). It is therefore reasonable to consider that an amino acid a is assigned to a vector whose components are the volume value $v(a)$ and the polarity value $p(a)$ introduced by Grantham (18). We use the normalized forms of these values: $\hat{v}(a) = [v(a) - \bar{v}]/\sigma_v$ and $\hat{p}(a) = [p(a) - \bar{p}]/\sigma_p$, where an overbar denotes the average over 20 amino acids, and σ_v and σ_p denote the standard deviations of volume and polarity, respectively. An amino acid sequence is converted to a sequence of such vectors.

MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform (2002)



MAFFT (multiple alignment using fast Fourier transform)

Calculation of the correlation between two amino acid sequences. We define the correlation $c(k)$ between two sequences of such vectors as

$$c(k) = c_v(k) + c_p(k), \quad 1$$

where $c_v(k)$ and $c_p(k)$ are, as defined below, the correlations of volume component and polarity component, respectively, between two amino acid sequences to be aligned. The correlation $c(k)$ represents the degree of similarity of two sequences with the positional lag of k sites. The high value of $c(k)$ indicates that the sequences may have homologous regions.

The correlation $c_v(k)$ of volume component between sequence 1 and sequence 2 with the positional lag of k sites is defined as

$$c_v(k) = \sum_{1 \leq n \leq N, 1 \leq n+k \leq M} \hat{v}_1(n) \hat{v}_2(n+k), \quad 2$$

where $\hat{v}_1(n)$ and $\hat{v}_2(n)$ are the volume component of the n th site of sequence 1 with the length of N and that of sequence 2 with the length of M , respectively. Considering $N \simeq M$ in many cases, equation 2 takes $O(N^2)$ operations. The FFT reduces the CPU time of this calculation to $O(N \log N)$ (19). If $V_1(m)$ and $V_2(m)$ are the Fourier transform of $\hat{v}_1(n)$ and $\hat{v}_2(n)$, i.e.

$$\hat{v}_1(n) \Leftrightarrow V_1(m) \quad 3$$

$$\hat{v}_2(n) \Leftrightarrow V_2(m), \quad 4$$

it is known that the correlation $c_v(k)$ is expressed as

$$c_v(k) \Leftrightarrow V_1^*(m) \cdot V_2(m), \quad 5$$

where $\Leftrightarrow$ represents transform pairs, and the asterisk denotes complex conjugation.

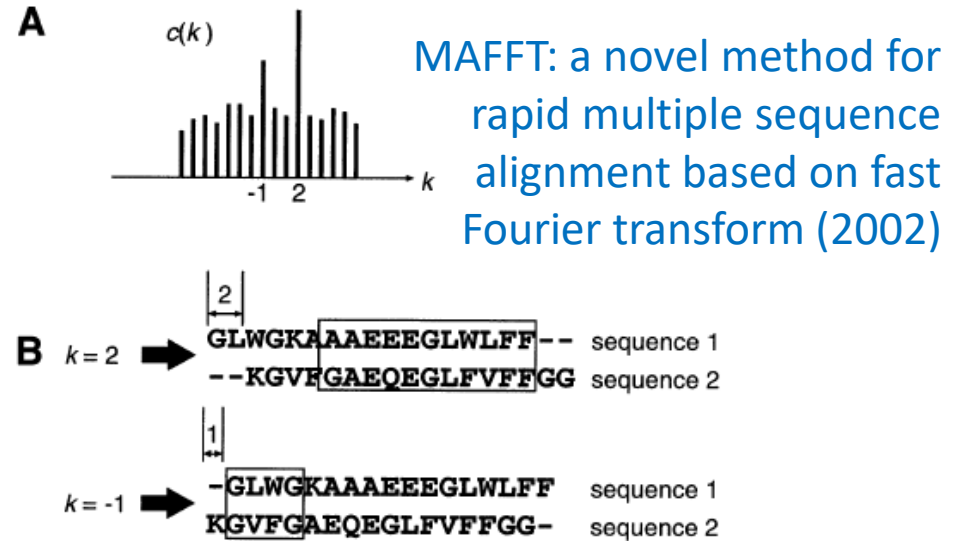


Figure 1. (A) A result of the FFT analysis. There are two peaks corresponding to two homologous blocks. (B) Sliding window analysis is carried out and the positions of homologous blocks are determined. Note that window size is 30 (see text) but the window size is set to 4 in (B) for simplicity.

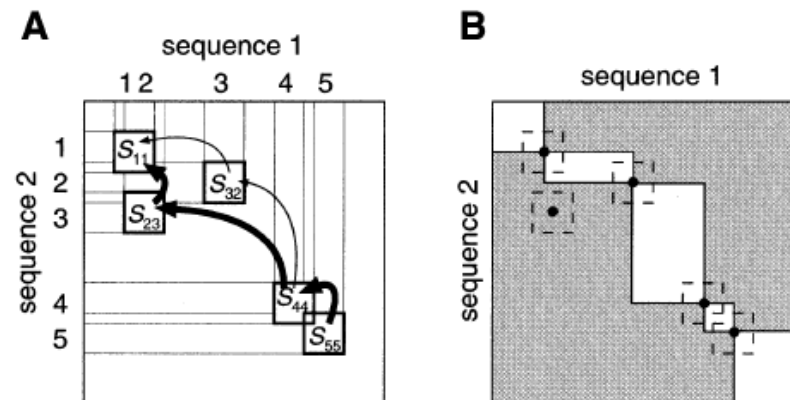


Figure 2. (A) An example of the segment-level DP; (B) Reducing the area for DP on a homology matrix.

Дискретное преобразование Фурье (DFT)

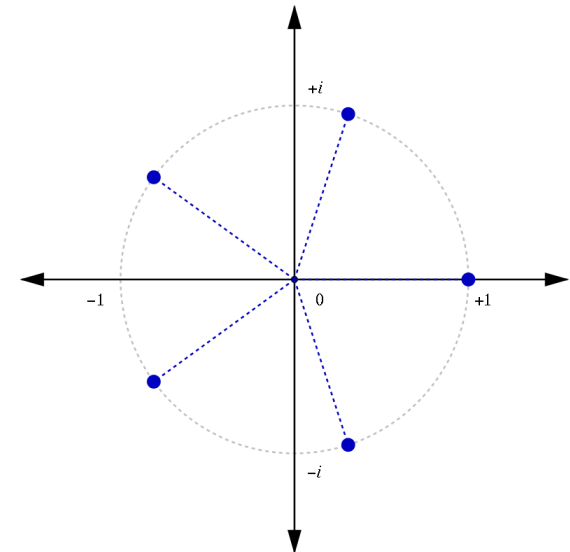
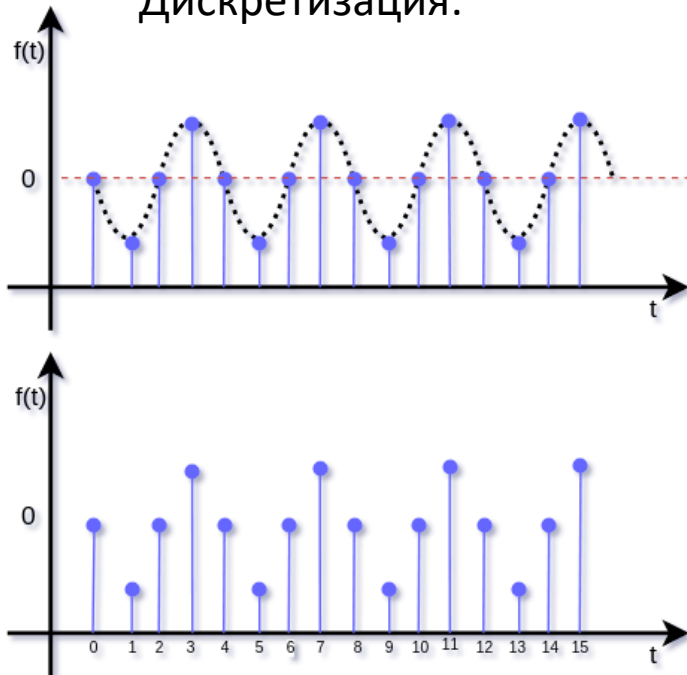
Прямое преобразование:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} = \sum_{n=0}^{N-1} x_n \cdot [\cos(2\pi kn/N) - i \cdot \sin(2\pi kn/N)], \quad (k = 0, \dots, N-1).$$

Обратное преобразование:

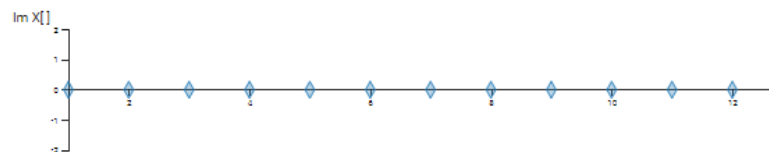
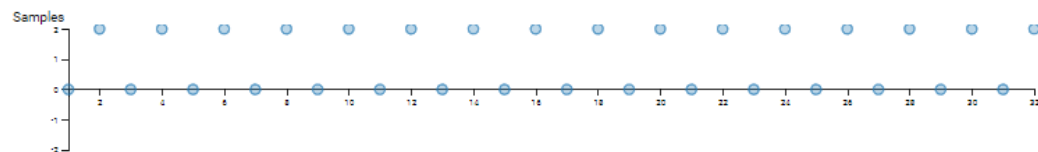
$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{\frac{2\pi i}{N} kn} = \frac{1}{N} \sum_{k=0}^{N-1} X_k \cdot [\cos(2\pi kn/N) + i \cdot \sin(2\pi kn/N)], \quad (n = 0, \dots, N-1).$$

Дискретизация:



$$\exp\left(\frac{2k\pi i}{n}\right) = \cos \frac{2k\pi}{n} + i \sin \frac{2k\pi}{n}, \quad k = 0, 1, \dots, n-1.$$

The Discrete Fourier Transform Sandbox



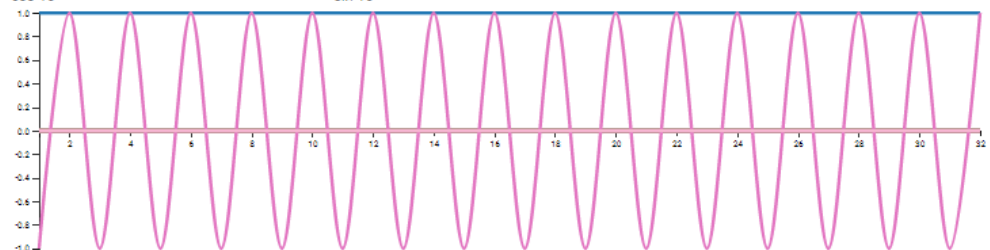
Synthesis (Cos + Sin)

cos 0
 cos 2
 cos 4
 cos 6
 cos 8
 cos 10
 cos 12
 cos 14
 cos 16

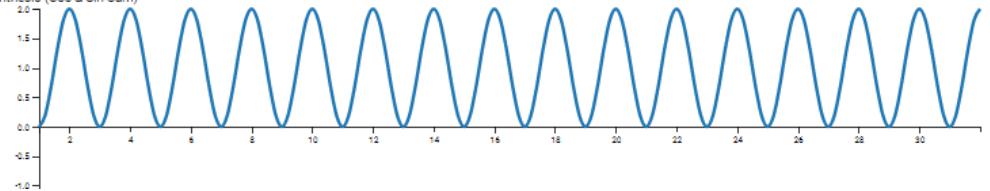
sin 0
 sin 2
 sin 4
 sin 6
 sin 8
 sin 10
 sin 12
 sin 14
 sin 16

cos 1
 cos 3
 cos 5
 cos 7
 cos 9
 cos 11
 cos 13
 cos 15

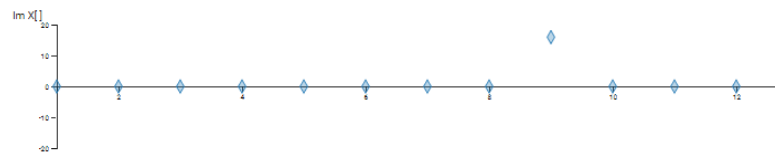
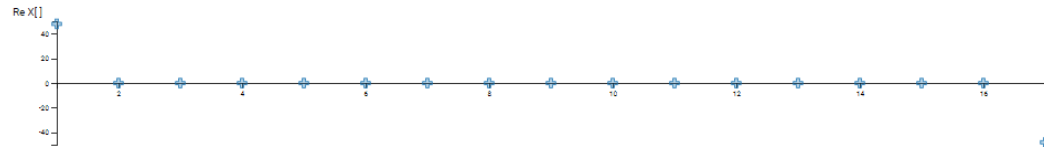
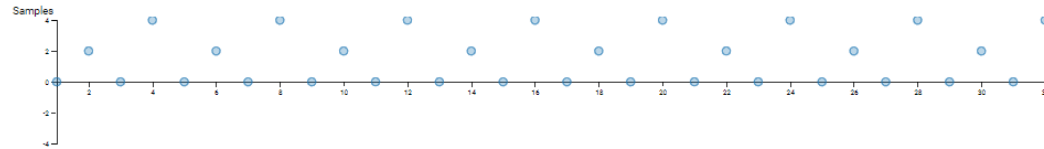
sin 1
 sin 3
 sin 5
 sin 7
 sin 9
 sin 11
 sin 13
 sin 15



Synthesis (Cos & Sin Sum)



The Discrete Fourier Transform Sandbox



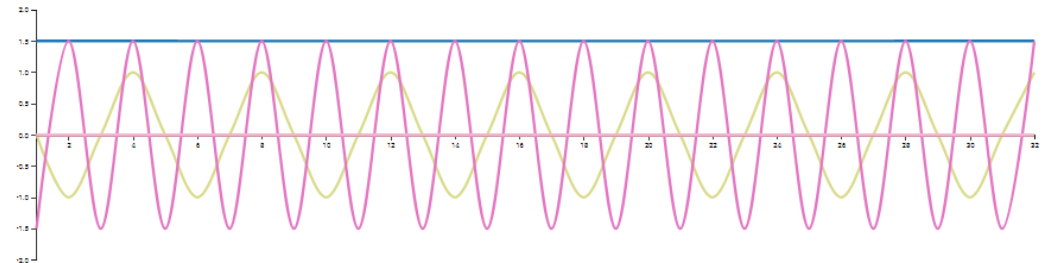
Synthesis (Cos + Sin)

- cos 0
- cos 2
- cos 4
- cos 6
- cos 8
- cos 10
- cos 12
- cos 14
- cos 16

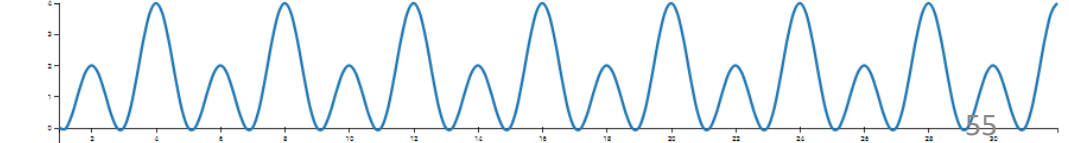
- sin 0
- sin 2
- sin 4
- sin 6
- sin 8
- sin 10
- sin 12
- sin 14
- sin 16

- cos 1
- cos 3
- cos 5
- cos 7
- cos 9
- cos 11
- cos 13
- cos 15

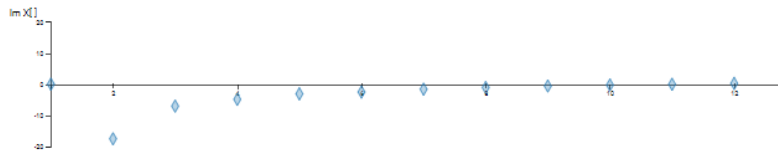
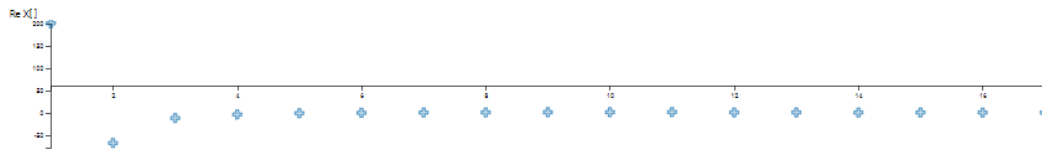
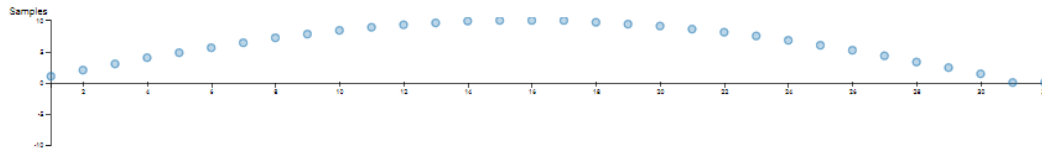
- sin 1
- sin 3
- sin 5
- sin 7
- sin 9
- sin 11
- sin 13
- sin 15



Synthesis (Cos & Sin Sum)



The Discrete Fourier Transform Sandbox



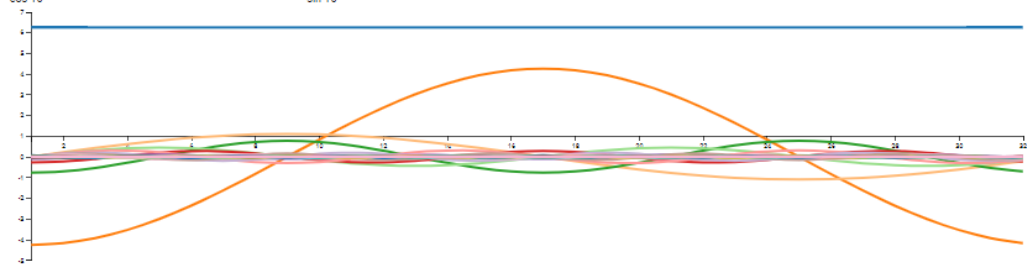
Synthesis (Cos + Sin)

- cos 0
- cos 2
- cos 4
- cos 6
- cos 8
- cos 10
- cos 12
- cos 14
- cos 16

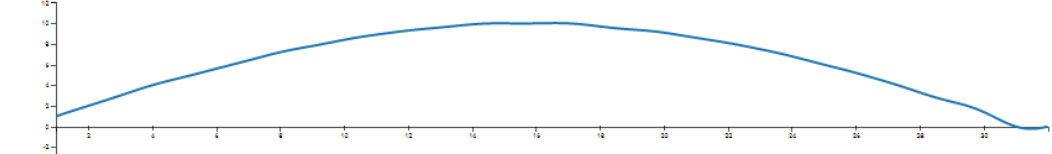
- sin 0
- sin 2
- sin 4
- sin 6
- sin 8
- sin 10
- sin 12
- sin 14
- sin 16

- cos 1
- cos 3
- cos 5
- cos 7
- cos 9
- cos 11
- cos 13
- cos 15

- sin 1
- sin 3
- sin 5
- sin 7
- sin 9
- sin 11
- sin 13
- sin 15



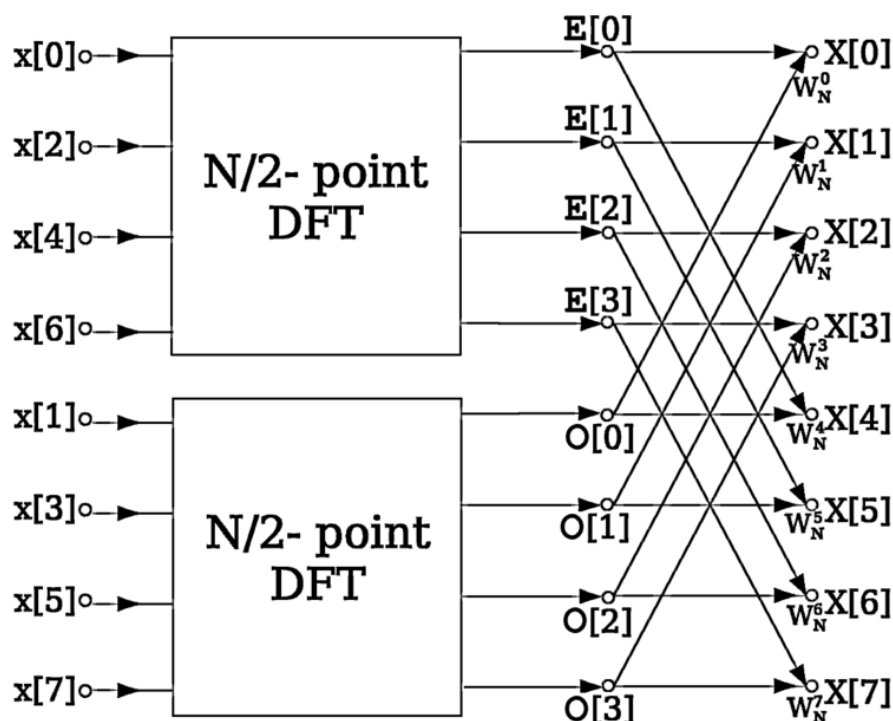
Synthesis (Cos & Sin Sum)



Быстрое преобразование Фурье (FFT)

(Gauss, 1805; Cooley, Tukey, 1965)

In 1994, Gilbert Strang described the FFT as "the most important numerical algorithm of our lifetime", and it was included in Top 10 Algorithms of 20th Century...



Сложность уменьшается с $O(N^2)$ до $O(N \log N)$ за счёт повторного использования промежуточных результатов

Сравнение алгоритмов выравнивания

<https://www.ebi.ac.uk/Tools/msa/>

Clustal Omega

New MSA tool that uses seeded guide trees and HMM profile-profile techniques to generate alignments. Suitable for medium-large alignments.

 Launch Clustal Omega
.....

EMBOSS Cons

EMBOSS Cons creates a consensus sequence from a protein or nucleotide multiple alignment.

 Launch EMBOSS Cons
.....

Kalign

Very fast MSA tool that concentrates on local regions. Suitable for large alignments.

 Launch Kalign
.....

MAFFT

MSA tool that uses Fast Fourier Transforms. Suitable for medium-large alignments.

 Launch MAFFT
.....

MUSCLE

Accurate MSA tool, especially good with proteins. Suitable for medium alignments.

 Launch MUSCLE
.....

+ несколько других

Сравнение алгоритмов выравнивания

Results for job clustalo-l20201015-000241-0473-47840372-p2m

Alignments

Result Summary

Guide Tree

Phylogenetic Tree

Results Viewers

Submission Details

Download Alignment File

Show Colors

CLUSTAL O(1.2.4) multiple sequence alignment

```

1MBN_1|Chain  -VLSEGEWQLVLHYNWAKVEADVAGHGQDILIRLFKSHPETLEKFDRFKHLKTEAEMKASE  59
1GDJ_1|Chain  GALTESQAALVKSWEEFNANIPKHTRFFILVLEIAPAAKDLFS-FLKG-TSEVPQNNP  58
1B0B_1|Chain  -SLSAAQKDNVKSSWAKASAANGTAGPEFFMALFDAHDDVFAKFSGLFSGAAKGTVKNT  59
      *:  .:  *  *  .:.*      :::  ::.  .  *.  :  :.  :.

1MBN_1|Chain  DLKKHGVTVLTALG--AILKKKGH--HEAELKPLAQSHATKHKIP---IKYLE
1GDJ_1|Chain  ELQAHAGKVFKLVYAAAIQLEVTGVVTDATLKNLGSVHVSKGVA----DAHFP
1B0B_1|Chain  EMAAQASQSFKGLVSNWVDNLDNAGALE--GQCKTFAANHKARGISAGQLEAAFKY
      ::  :.  .:  :  *  *  .  *  :.  *  ::  :

1MBN_1|Chain  IHVLHSRHPGDFGADAQGAMNK----ALELFRKDIAAKYKELGYQG
1GDJ_1|Chain  LKTIKEVVGAKWSEELNSAWTIAYDELAIVIKKEMDDAA-----
1B0B_1|Chain  --FMKSY-----GGDEGANTAVAGALMGMIKPD-----
      :.:  :.:  .  :.:  :.
    
```

This is a Neighbour-joining tree without distance corrections.

Branch length: ☐ Cladogram ☒ Real



1MBN_1|Chain 0.43847
1GDJ_1|Chain 0.36573
1B0B_1|Chain 0.37692

Tree Data

```

(
  1MBN_1|Chain:0.43847,
  1GDJ_1|Chain:0.36573,
  1B0B_1|Chain:0.37692);
    
```

```

#
# Percent Identity Matrix
# - created by Clustal2.1
#
1: 1MBN_1|Chain 100.00 19.58 18.46
2: 1GDJ_1|Chain 19.58 100.00 25.74
3: 1B0B_1|Chain 18.46 25.74 100.00
    
```

Сравнение алгоритмов выравнивания. MAFFT, MUSCLE

CLUSTAL format alignment by MAFFT FFT-NS-i (v7.429)

```
1GDJ_1|Chain  GALTESQAALVKSSWEEFNANIPKHTRFFILVLEIAPAAKDLFSFL-----KGTSEVPQ
1MBN_1|Chain  -VLSEGEWQLVLHVWAKVEADVAGHGQDILIRLFKSHPETLEKFD RF-----
1B0B_1|Chain  -SLSAAQKDNVKSSWAKASAAWGTAGPEFFMALFDAHDDVFAKFSGLFSGAAKGTV---K
               *:  .:  *  *  :  .*               :::  ::.  .  *.  :

1GDJ_1|Chain  NNPELQA HAGKVFKLVYEAAIQLEVTGVVVT-----DATLKNLGSVHVSK-GVADAHFPV
1MBN_1|Chain  KHLKTEAEMKASEDLKKHGVTVLTALGAILKKKGHEAELKPLAQSHATKHKIPIKYLEF
1B0B_1|Chain  NTPEMAAQAQSFKGLVSNWVDNLDNAGAL-----EGQCKTFAANH KAR-GISAGQLEA
               :  :  *  *  .  *  *  .:  :.  *  :.  *  ::  :.  :

1GDJ_1|Chain  VKEAILKTIKEV-VGAKWSE---ELNSAWTIAYDELAIVIKKEMDDAA
1MBN_1|Chain  ISEAI IHVLSRHPGDFGADAQGAMNKALELFRKDIAAKYKELGY-QG
1B0B_1|Chain  AFKVLAGFMKSY-GGDEG-----AWTAVAGALMGMI RPDM----
               :::  ::.  *  *  :  :
```

CLUSTAL multiple sequence alignment by MUSCLE (3.8)

```
1GDJ_1|Chain  GALTESQAALVKSSWEEFNANIPKHTRFFILVLEIAPAAKDLFS-FLKGTSEVP-QNNP
1MBN_1|Chain  -VLSEGEWQLVLHVWAKVEADVAGHGQDILIRLFKSHPETLEKFD RFKHLKTEAEMKASE
1B0B_1|Chain  -SLSAAQKDNVKSSWAKASAAWGTAGPEFFMALFDAHDDVFAKFSGLFSGAAKGTVKNT P
               *:  .:  *  *  :  .*               :::  ::.  .  *.  :  :.  :

1GDJ_1|Chain  ELQA HAGKVFKLVYEAAIQLEVTGVVVT DATLKNLGSVHVSK-GVADAHFPV VKEAILKT
1MBN_1|Chain  DLKKHGVTVLTALGAI---LKKKGHH--EAELKPLAQSHATKHKIPIKYLEF ISEAI IHV
1B0B_1|Chain  EMAAQAQSFKGLVSNWVDNLDNAGAL--EGQCKTFAANH KAR-GISAGQLEAAFKVLAGE
               ::  :.  ..  :  *  *  :.  *  :.  *  :.  :.  :.  ::

1GDJ_1|Chain  IKEVVGAKWSEELNSAWTIAYDELAIVIK---KEMDDAA
1MBN_1|Chain  LHSRHPGDFGADAQGAMNKALELFRKDIAAKYKELGYQG
1B0B_1|Chain  MKS-YGGD-----EGAWTAVAGALMGMI R---PDM----
               ::.  ..  :.*  .  :  *  ::
```

Сравнение алгоритмов выравнивания. ClustalO, structure

```

CLUSTAL O(1.2.4) multiple sequence alignment

1GDJ_1|Chain  GALTESQAALVKSSWEEFNANIPKHTHRFFILVLEIAPAAKDLFS-FLKG-TSEVPQNNP 58
1MBN_1|Chain  -VLSEGEWQLVLHVWAKVEADVAGHGQDILIRLFKSHPETLEKFDRFKHLKTEAEMKASE 59
1B0B_1|Chain  -SLSAAQKDNVKSWSAKASAAWGTAGPEFFMALFDAHDDVFAKFSGLFSGAAKGTVKNT 59
               *:  .:   *   *  :  .*           :::  ::.   .   *.  :   :.   :.

1GDJ_1|Chain  ELQAAGKVFVKLVYEAAIQLEVTGVVVTDATLKNLGSVHVSKGVA----DAHFPVVKEAI 114
1MBN_1|Chain  DLKKHGVTVLTALG---AILKKKGH--HEAELKPLAQSHATKHKIP---IKYLEFISEAI 111
1B0B_1|Chain  EMAAQASFKGLVSNWVDNLDNAGALE--GQCKTFAANHKGARGISAGQLEAAFKVL 115
               ::  :.  .:   :           *.   *   .   *  :.  *  ::           :  ::

1GDJ_1|Chain  LKTIKEVVGAKWSEELNSAWTIAYDELAIVIKKEMDDAA----- 153
1MBN_1|Chain  IHVLHSRHPGDFGADAQGAMNK----ALELFRKDIAAKYKELGYQG 153
1B0B_1|Chain  --FMKSY-----GGDEGAWTAVAGALMGMI RPD----- 142
               :.:           :.*  .           :::  ::
    
```

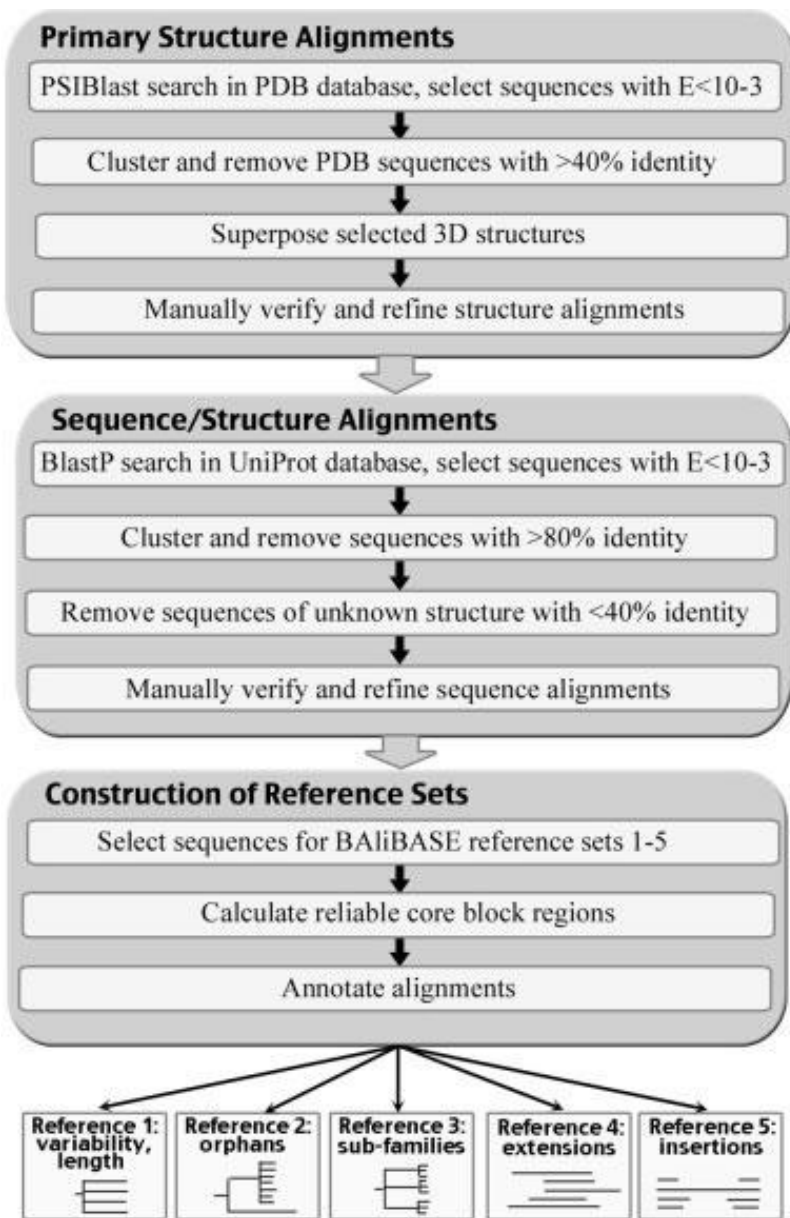
```

1GDJ      1  CCCC HHHHHHHHHHHHHHHHHHH C HHHHHHHHHHHHHHHHH C HHHHHHH CCCCCCCC--
1MBN      1  -CCC HHHHHHHHHHHHHHHHHHH C HHHHHHHHHHHHHHHHH C HHHHHHH CCCCCCCC HH
1GDJ      1  GALTESQAALVKSSWEEFNANIPKHTHRFFILVLEIAPAAKDLFSFLKGTSE--
1MBN      1  -VLSEGEWQLVLHVWAKVEADVAGHGQDILIRLFKSHPETLEKFDRFKHLKTEA
               .*:*.:.  **  .  *  :.:*::.  *  :.:*:  :.:  *  :  :  *.  :*  .

1GDJ     53  -CCCCC HHHHHHHHHHHHHHHHHHHHHHHHHHHHH CCCCCC HHHHHHHHHHHHHHHHH -CCCC
1MBN     54  HH-HHH CHHHHHHHHHHHHHHHHHHHHHHHHHHHH -C--CCC-- HHHHHHHHHHHHHHHHH CCCCC H
1GDJ     53  -VPQNNPELQAAGKVFVKLVYEAAIQLEVTGVVVTDATLKNLGSVHVSK-GVAD
1MBN     54  EM-KASEDLKKHGVTVLTALGAILKK-K--GHH--EAELKPLAQSHATKHKIPI
               :  :  .  :*:  *.  *:  :           :  *           :*  **  *  .  *.:*  :

1GDJ    105  HHHHHHHHHHHHHHHHHHHHH CCCCC HHHHHHHHHHHHHHHHHHHHHHHHHHH -H--
1MBN    102  HHHHHHHHHHHHHHHHHHHHH CCCCC HHHHHHHHHHHHHHHHHHHHHHHHHHH CCCC
1GDJ    105  AHFPVVKEAILKTIKEVVGAKWSEELNSAWTIAYDELAIVIKKEMDDA-A--
1MBN    102  KYLEFISEAIIHVLHSRHPGDFGADAQGAMNKALELFRKDIAAKYKELGYQG
               :.  :.***:.:.:.  .:.  :  :.*  *  :  :  *  :  :
    
```

Сравнение алгоритмов выравнивания

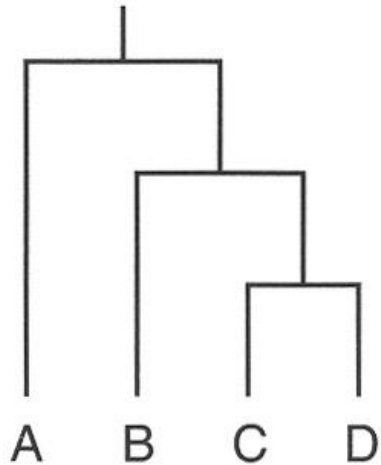


BALiBASE: a benchmark alignment database for the evaluation of multiple alignment programs (1999 - ...)

Assessing the efficiency of multiple sequence alignment programs (2014)

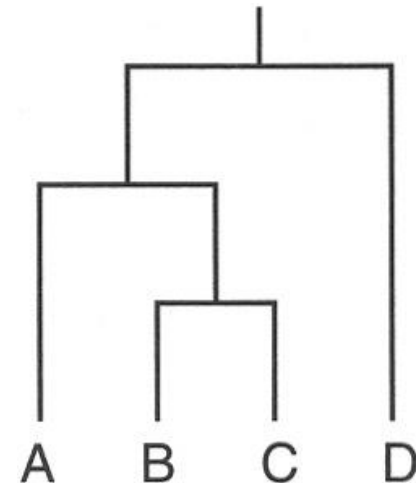
Results: Our results indicate that mostly the consistency-based programs Probcons, T-Coffee, Probalign and MAFFT outperformed the other programs in accuracy. Whenever sequences with large N/C terminal extensions were present in the BALiBASE suite, Probalign, MAFFT and also CLUSTAL OMEGA outperformed Probcons and T-Coffee. The drawback of these programs is that they are more memory-greedy and slower than POA, CLUSTALW, DIALIGN-TX, and MUSCLE. CLUSTALW and MUSCLE were the fastest programs, being CLUSTALW the least RAM memory demanding program.

Филогенетические деревья – проблема переменной скорости эволюции

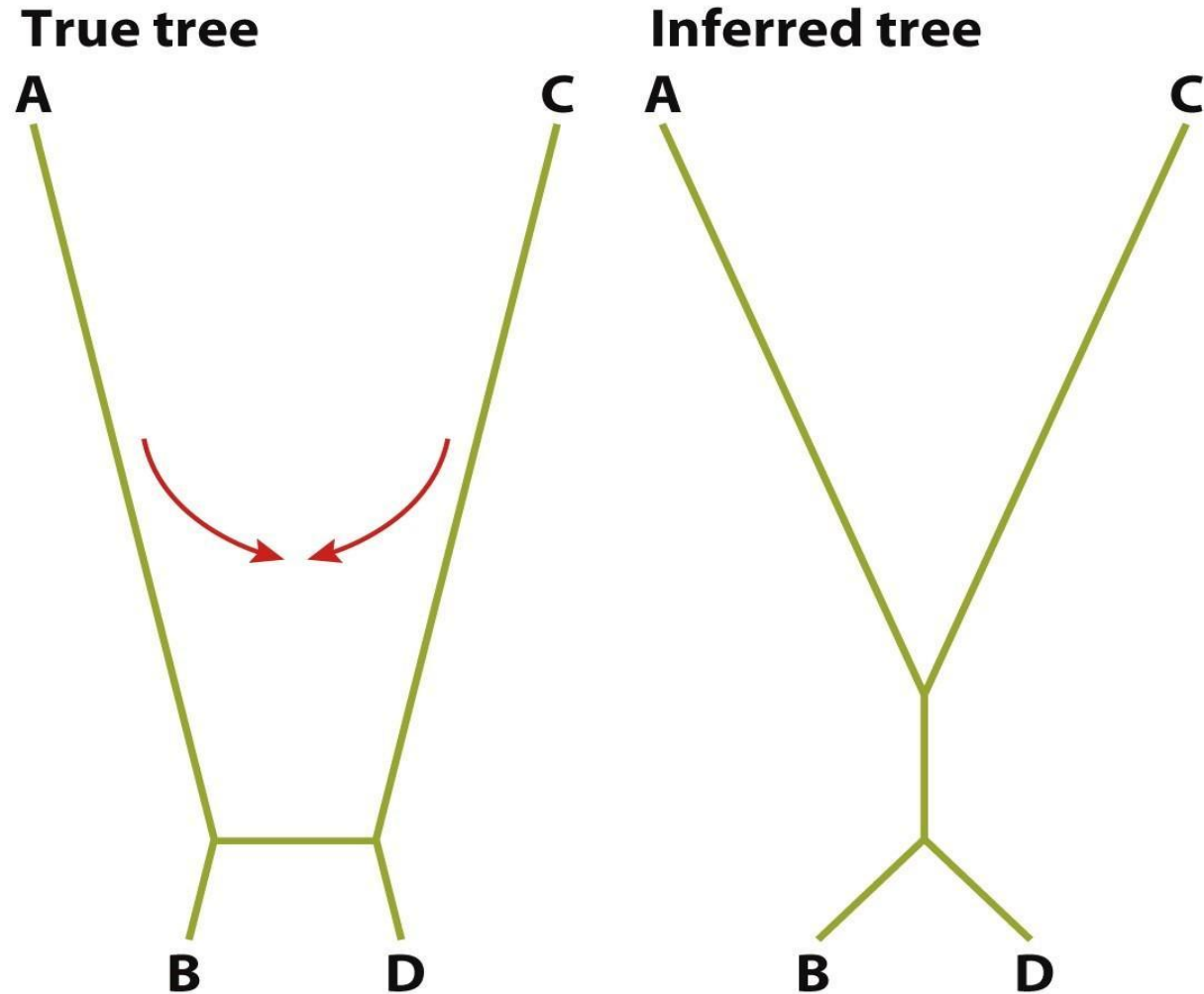


	A	B	C	D
A	0	3	3	3
B		0	2	2
C			0	1
D				0

	A	B	C	D
A	0	3	3	20
B		0	2	20
C			0	20
D				0



Филогенетические деревья – «притяжение длинных ветвей»

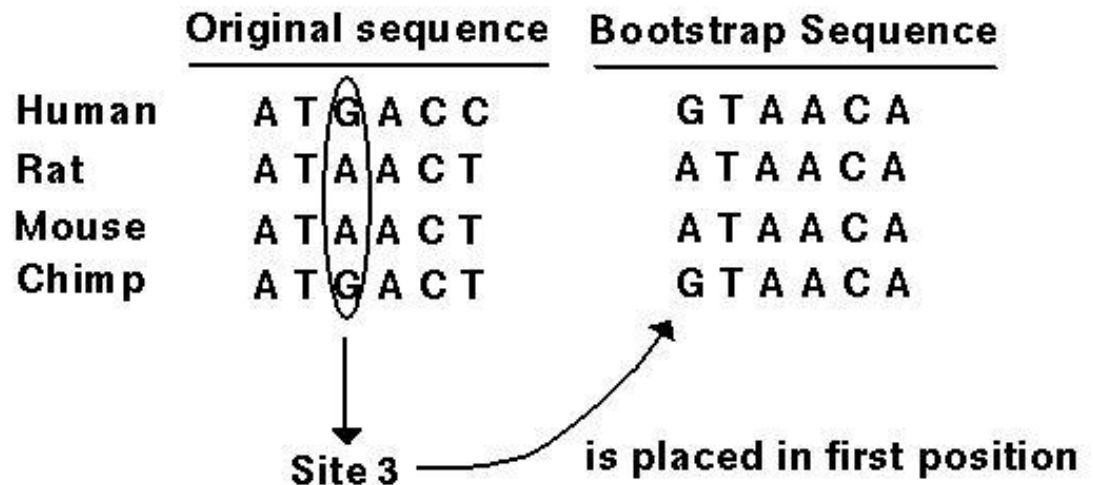


Филогенетические деревья – методы проверки

- 1) **Использование внешней группы**, т.е. видов, которые заведомо более удалены ото всех видов, для которых строится дерево (приматы и корова);
- 2) **Сравнение деревьев**, полученных на основе разных характеристик. Очевидно, они должны быть согласованными;

Оценка результата с помощью формальных статистических тестов:

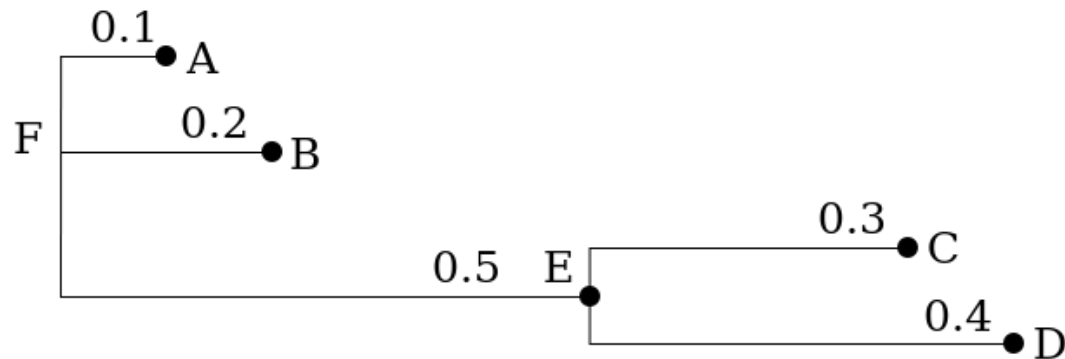
- 3) Например, **построение дерева для подмножества последовательностей** из исходного множественного выравнивания должно дать поддереву дерева, полученного для этого выравнивания (jack-knife);
- 4) **Бутстреп** (boot strap).



(Then the next five randomly chosen sites: 2, 1, 1, 5, 4, are placed in the next five positions.)

Филогенетические деревья.

Скобочная формула (Newick format) (1986)



<code>(, , (,)) ;</code>	<i>имена узлов не указаны</i>
<code>(A , B , (C , D)) ;</code>	<i>указаны только имена листьев</i>
<code>(A , B , (C , D) E) F ;</code>	<i>указаны имена всех узлов</i>
<code>(: 0.1 , : 0.2 , (: 0.3 , : 0.4) : 0.5) ;</code>	<i>для всех узлов кроме корня указано расстояние до родительского узла</i>
<code>(: 0.1 , : 0.2 , (: 0.3 , : 0.4) : 0.5) : 0.0 ;</code>	<i>для всех узлов указано расстояние до родительского узла</i>
<code>(A : 0.1 , B : 0.2 , (C : 0.3 , D : 0.4) : 0.5) ;</code>	<i>указаны имена листьев и расстояния</i>
<code>(A : 0.1 , B : 0.2 , (C : 0.3 , D : 0.4) E : 0.5) F ;</code>	<i>указаны все имена и расстояния</i> 67

Модели эволюции. Модель Джукса-Кантора (1969)

Пусть вероятность α замены любого нуклеотида на любой другой в единицу времени одинакова.

$P_A(0) = 1$ – в некой позиции в момент времени $t = 0$ содержится нуклеотид A

$P_A(1) = 1 - 3\alpha$ – вероятность сохранения нуклеотида A в этой позиции в момент $t = 1$

$P_A(2) = (1 - 3\alpha)P_A(1) + \alpha(1 - P_A(1))$ – вероятность сохранения нуклеотида A в этой позиции в момент $t = 2$.

Для произвольного момента t аналогично:

$$P_A(t+1) = (1 - 3\alpha)P_A(t) + \alpha(1 - P_A(t))$$

$$\Delta P_A(t) = P_A(t+1) - P_A(t) = -3\alpha P_A(t) + \alpha(1 - P_A(t)) = -4\alpha P_A(t) + \alpha$$

Переходя к непрерывному времени, получаем

$$d P_A(t)/dt = -4\alpha P_A(t) + \alpha$$

- дифференциальное уравнение, решение которого можно искать в виде

$$P_A(t) = U \exp (-4\alpha t) + V.$$

Модели эволюции. Модель Джукса-Кантора (1969)

Подставляя, находим $V = \frac{1}{4}$, что с учетом известного $P_A(0)$ даёт

$$P_A(t) = (P_A(0) - \frac{1}{4}) \exp(-4\alpha t) + \frac{1}{4}.$$

Если в момент 0 нуклеотид А присутствовал в позиции, то $P_A(0) = 1$ и

$$P_A(t) = \frac{3}{4} \exp(-4\alpha t) + \frac{1}{4}.$$

В противоположном случае $P_A(0) = 0$ и

$$P_A(t) = -\frac{1}{4} \exp(-4\alpha t) + \frac{1}{4}.$$

Таким образом, для любого нуклеотида при любом начальном состоянии со временем вероятность обнаружения этого нуклеотида в данной позиции стремиться к $\frac{1}{4}$, а значит доля нуклеотида в последовательности к 25% - **в общем случае это неверно**

Заметим, что эти же формулы справедливы и вообще для любых замен:

$$P_{ij}(t) = \frac{3}{4} \exp(-4\alpha t) + \frac{1}{4}.$$

$$P_{ij}(t) = -\frac{1}{4} \exp(-4\alpha t) + \frac{1}{4}.$$

Модели эволюции. Модель Джукса-Кантора (1969)

Рассмотрим теперь две гомологичные последовательности $\mathcal{X}$ и $\mathcal{Y}$ имеющие неизвестного общего предка.

Если в некой позиции этих последовательностей наблюдается нуклеотид А, то он мог остаться от общего предка с вероятностью $P_{AA}(t)*P_{AA}(t)$, либо в обеих последовательностях произошли одинаковые замены, вероятность чего

$$P_{CA}(t)*P_{CA}(t) + P_{GA}(t)*P_{GA}(t) + P_{TA}(t)*P_{TA}(t).$$

Итого нуклеотид А будет наблюдаться в обеих последовательностях с вероятностью

$$P_{AA}(t)^2 + P_{CA}(t)^2 + P_{GA}(t)^2 + P_{TA}(t)^2.$$

Это же верно и для всех позиций в целом, поэтому доля $I(t)$ идентичных нуклеотидов в последовательностях $\mathcal{X}$ и $\mathcal{Y}$ будет

$$I(t) = P_{AA}(t)^2 + P_{CA}(t)^2 + P_{GA}(t)^2 + P_{TA}(t)^2 = \frac{3}{4} \exp(-8\alpha t) + \frac{1}{4}$$

Модели эволюции. Модель Джукса-Кантора (1969)

При этом доступная для наблюдения доля p различий между последовательностями $\mathcal{X}$ и $\mathcal{Y}$ будет

$$p = 1 - I(t) = \frac{3}{4} (1 - \exp(-8\alpha t)),$$

откуда для αt имеем

$$8\alpha t = -\ln(1 - 4/3p).$$

При этом расчетное число замен в каждой из последовательностей за это время, согласно модели, составит $3\alpha t$, а расстояние между последовательностями

$$D = 2(3\alpha t).$$

Выражая αt и подставляя в предыдущее равенство, получаем

$$D = -\frac{3}{2} \ln(1 - 4/3p).$$

Другие модели (Кимуры, Таджимы-Неи и др.) в разной степени учитывают различия частот встречаемости нуклеотидов и частот их замен.

Расстояние с поправкой Пуассона

$p = n_d/n$ – наблюдаемая доля замен при сравнении последовательностей.

r – скорость замен **остатков**, одинаковая для всех позиций

rt – число замен за время t

Есть основания полагать, что вероятность k замен в одной и той же позиции подчиняется распределению Пуассона:

$$P(k; t) = \frac{(rt)^k}{k!} e^{-rt}$$

Для консервативных позиций по определению $k = 0$ и $P(0; t) = e^{-rt}$, а число таких позиций в последовательности длины n будет ne^{-rt} .

Вероятность того, что позиция осталась консервативной в обеих дочерних последовательностях будет

$$q = (e^{-rt})^2 = e^{-2rt}$$

Для оценки можно считать, что $q = (1 - p)$, тогда для расстояния между последовательностями $d = 2rt$ получаем

$$d = -\ln(1-p)$$

Molecular Evolutionary Genetics Analysis (1993)

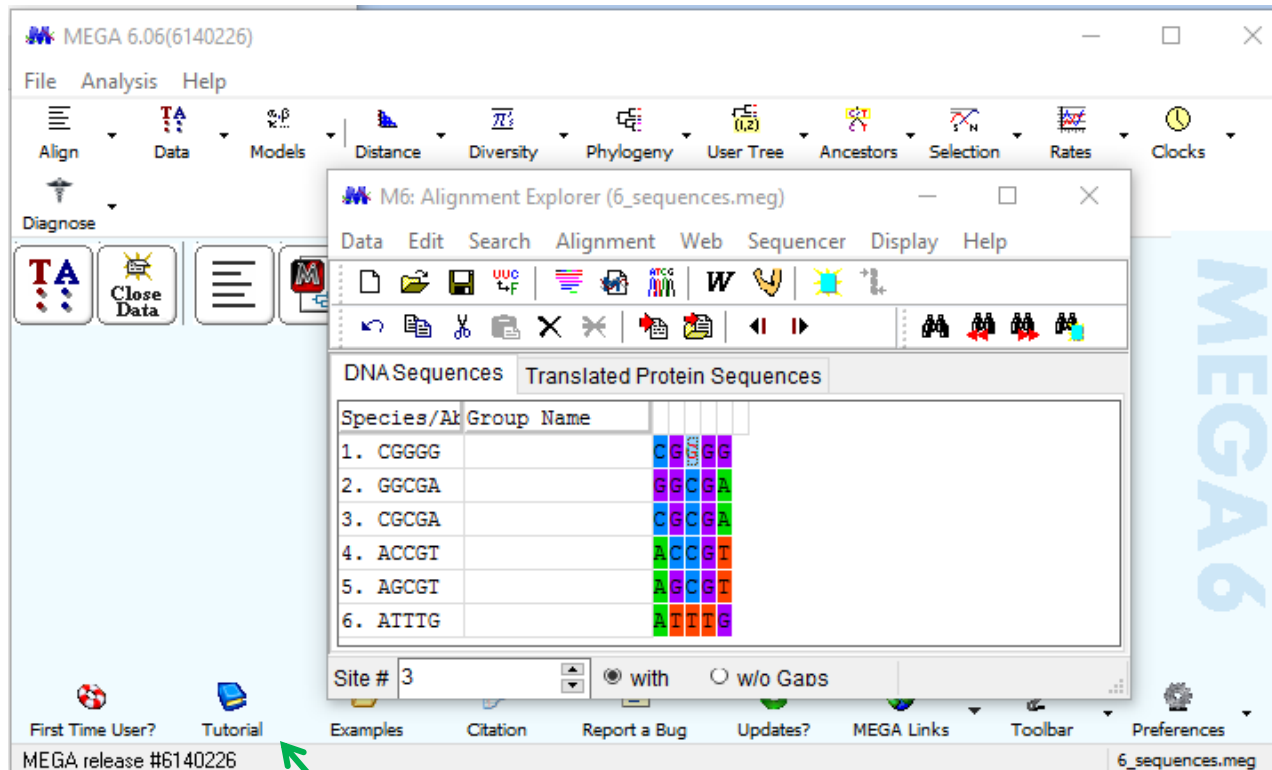


Molecular Evolutionary Genetics Analysis



Масатоси Неи

EN = 4



Tutorial (in English)

Molecular Evolutionary Genetics Analysis (1993)

M6: Analysis Preferences

Options Summary

Option	Selection
Analysis	Phylogeny Reconstruction
Scope	All Selected Taxa
Statistical Method	Neighbor-joining
Phylogeny Test	
Test of Phylogeny	Bootstrap method
No. of Bootstrap Replications	100
Substitution Model	
Substitutions Type	Nucleotide
Genetic Code Table	Not Applicable
Model/Method	No. of differences
Fixed Transition/Transversion Ratio	No. of differences
Substitutions to Include	p-distance Jukes-Cantor model Kimura 2-parameter model Tajima-Nei model Tamura 3-parameter model Tamura-Nei model Maximum Composite Likelihood
Rates and Patterns	
Rates among Sites	
Gamma Parameter	
Pattern among Lineages	Same (Homogeneous)
Data Subset to Use	
Gaps/Missing Data Treatment	Pairwise deletion
Site Coverage Cutoff (%)	Not Applicable
Select Codon Positions	☒ 1st ☒ 2nd ☒ 3rd ☒ Noncoding Sites

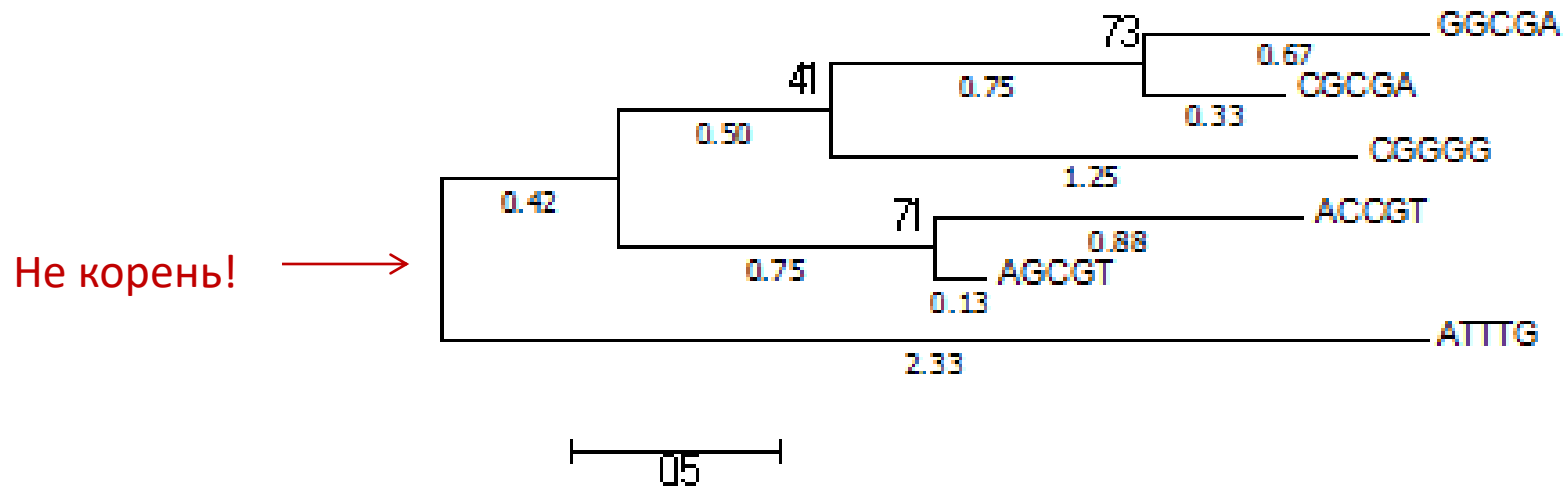
? Help ☒ Compute ☒ Cancel

Molecular Evolutionary Genetics Analysis (1993)

CGGGG	CGGGG
GGCGA	GGCGA
CGCGA	CGCGA
ACCGT	ACCGT
AGCGT	AGCGT
ATTG	ATTG

This is a *Neighbour-joining* tree.

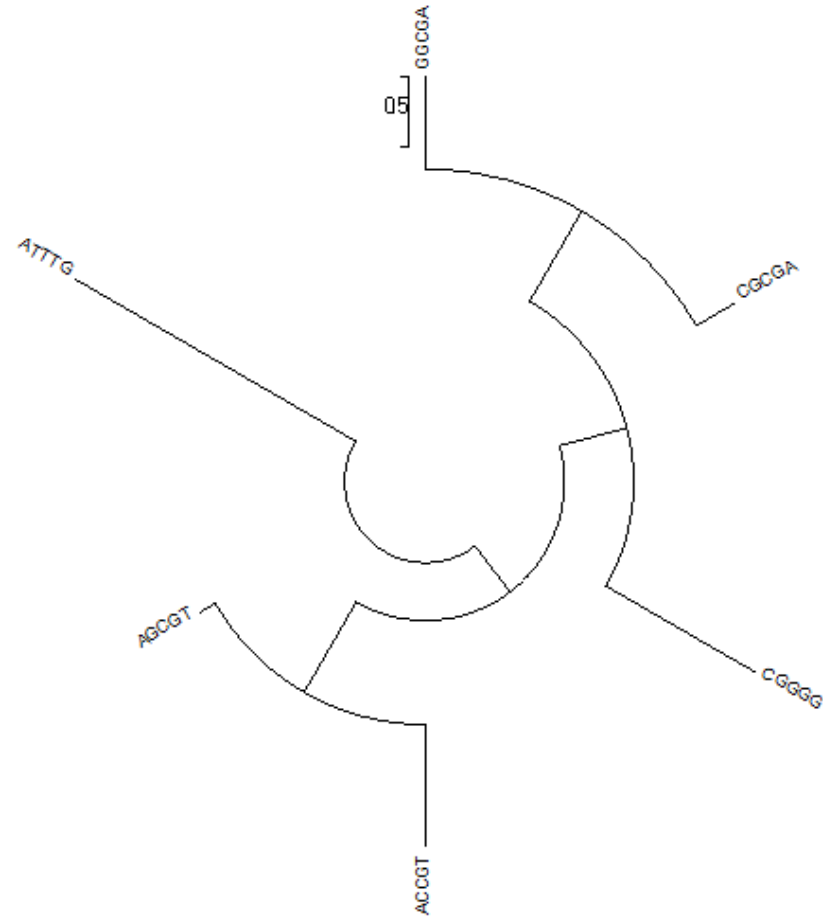
The evolutionary distances were computed using the *number of differences* method.



```
((((GGCGA:0.667,CGCGA:0.333):0.750,CGGGG:1.250):0.500,
(ACCGT:0.875,AGCGT:0.125):0.750):0.417,ATTG:2.333)
```

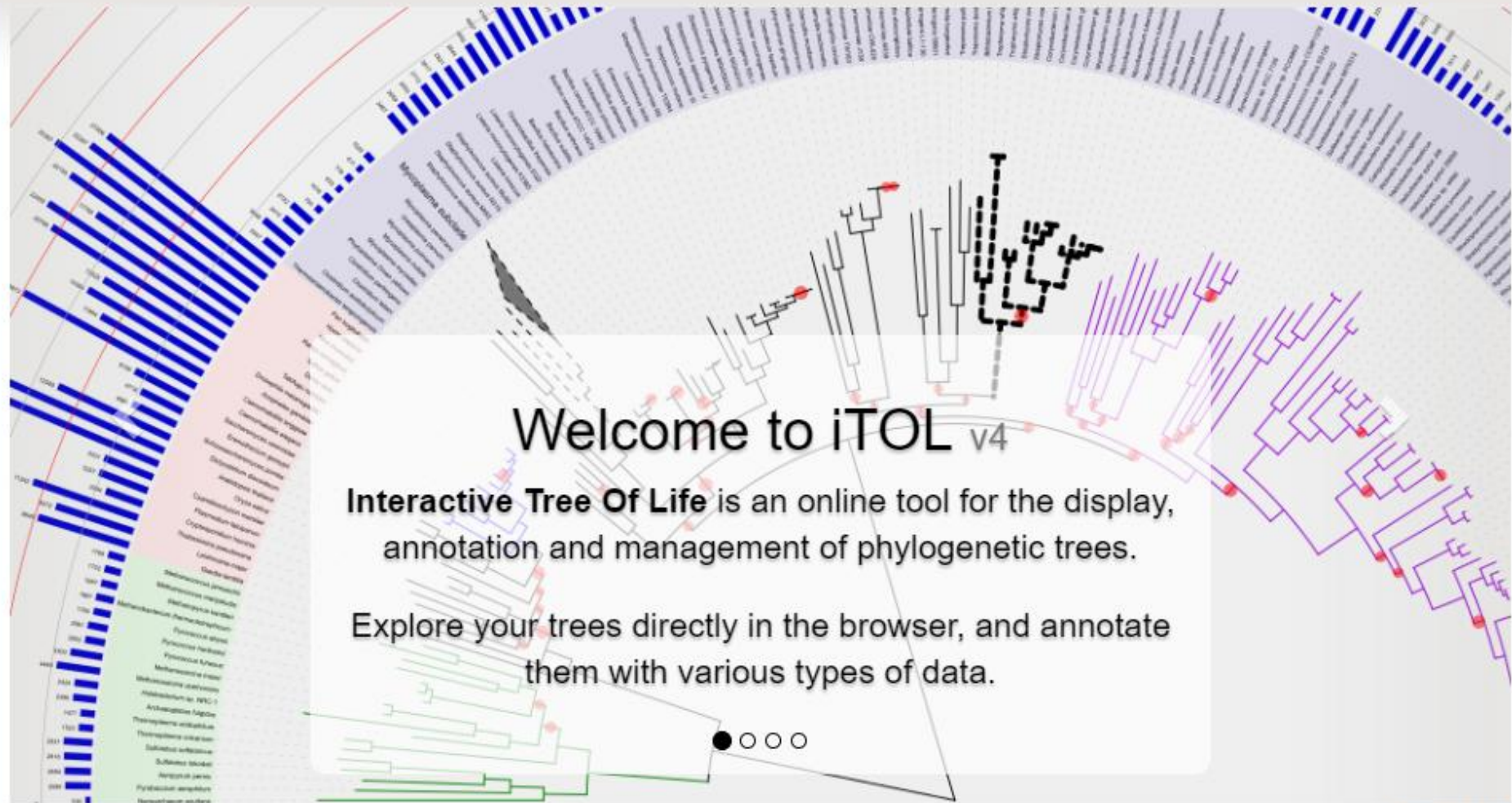

Molecular Evolutionary Genetics Analysis (1993)

CGGGG	CGGGG
GGCGA	GGCGA
CGCGA	CGCGA
ACCGT	ACCGT
AGCGT	AGCGT
ATTTG	ATTTG

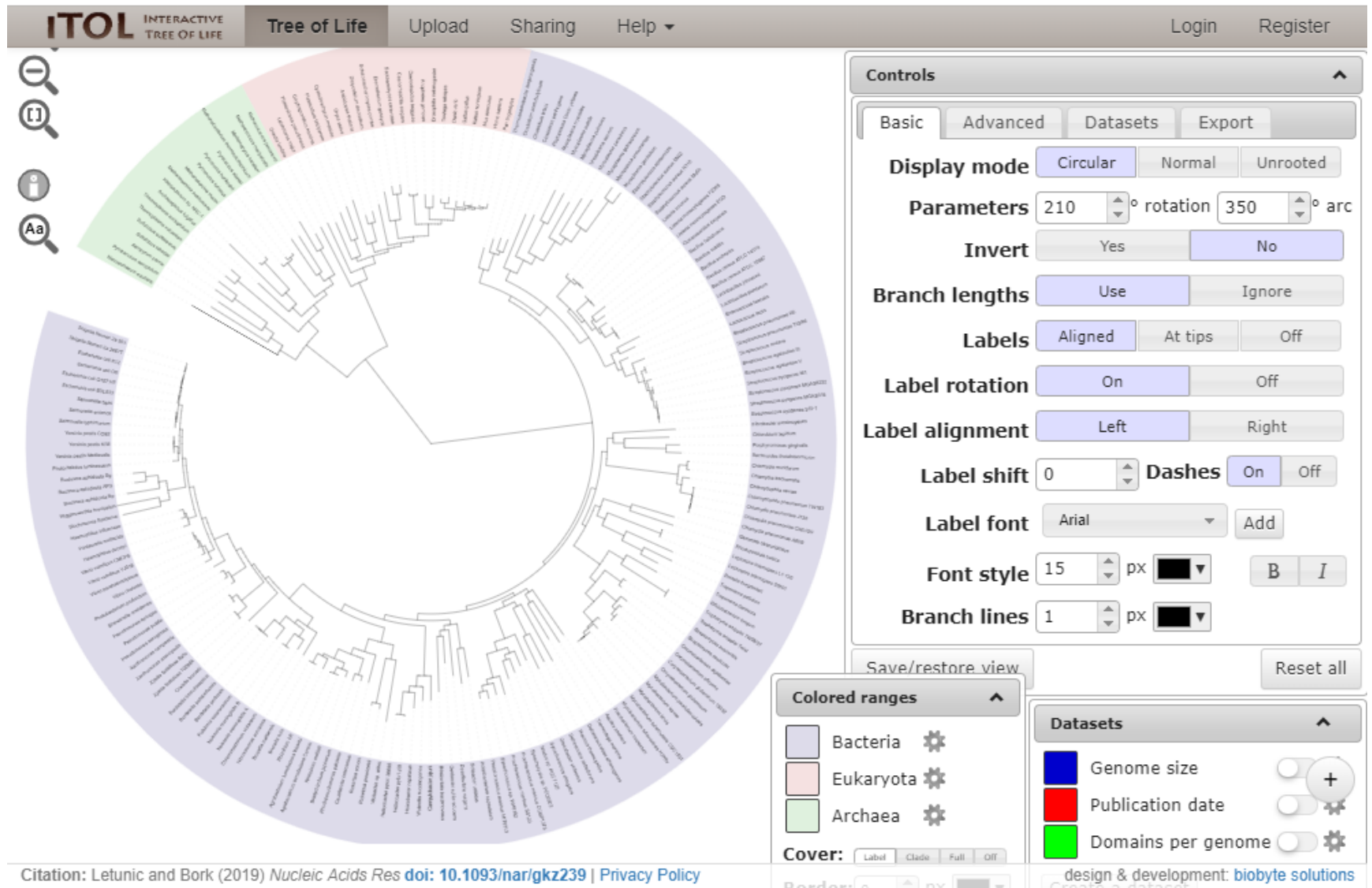


((((GGCGA:0.667,CGCGA:0.333):0.750,CGGGG:1.250):0.500,
 (ACCGT:0.875,AGCGT:0.125):0.750):0.417,ATTTG:2.333)

Interactive Tree of Life



Interactive Tree of Life



Множественное выравнивание последовательностей

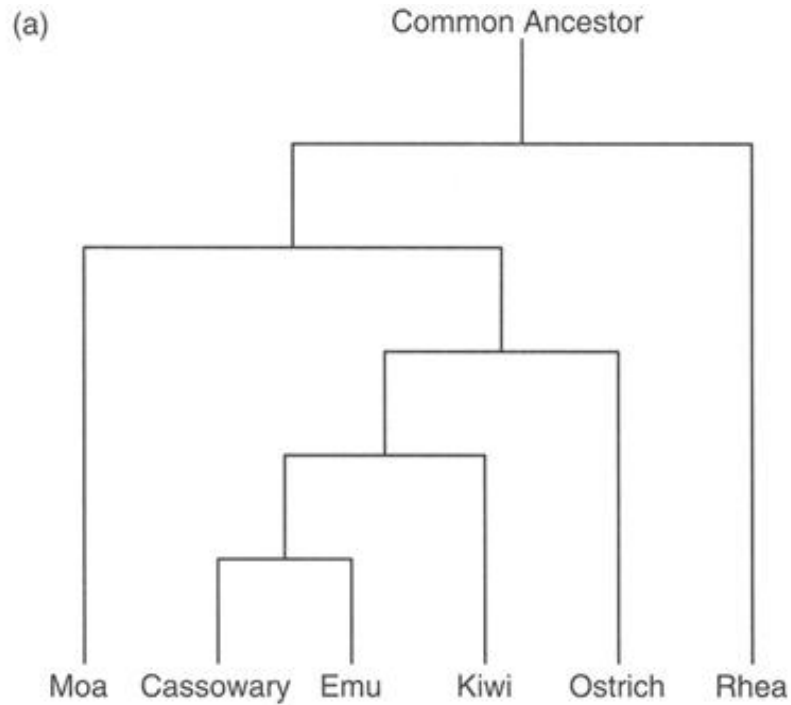
Цели:

- Построение филогенетических деревьев
- Выявление консервативных остатков и мотивов
- Построение профилей (визуализация)
- Итеративное выявление удаленной гомологии
- ...

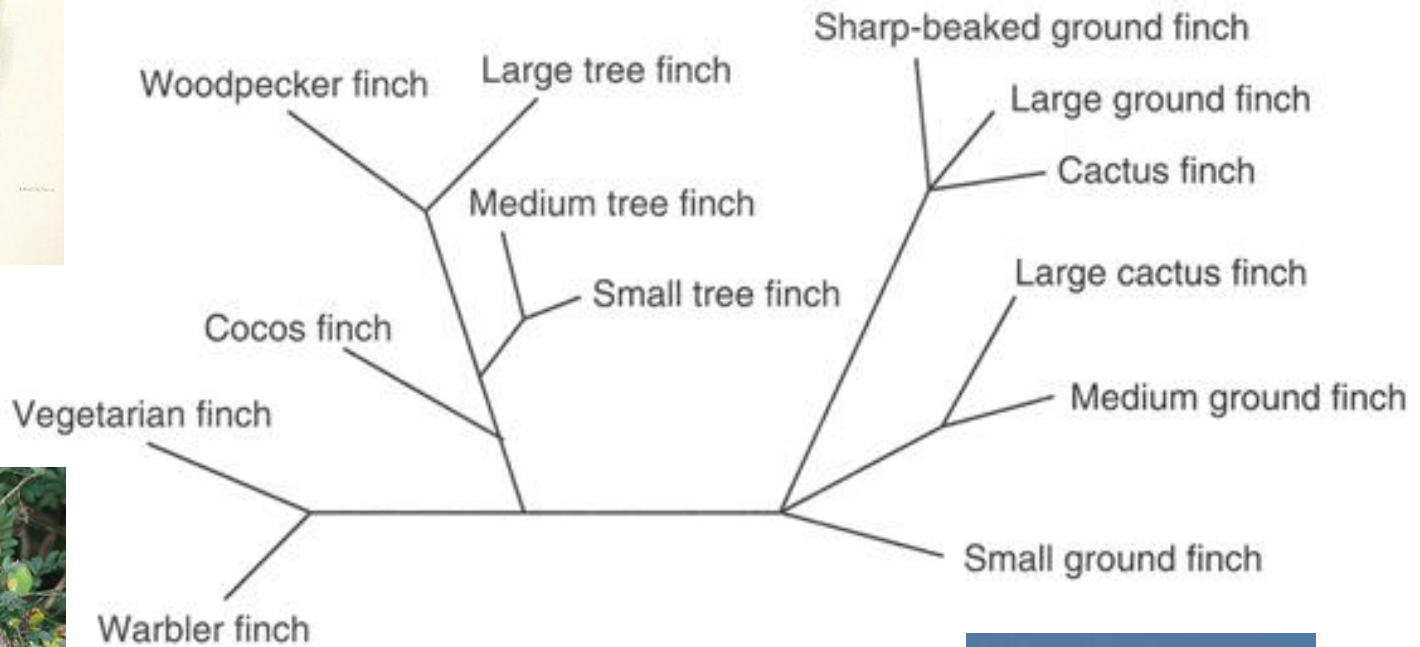
Алгоритмы:

- Динамическое программирование – не годится
- Прогрессивное выравнивание
- Скрытые марковские модели – в следующий раз
- Квантовые компьютеры?

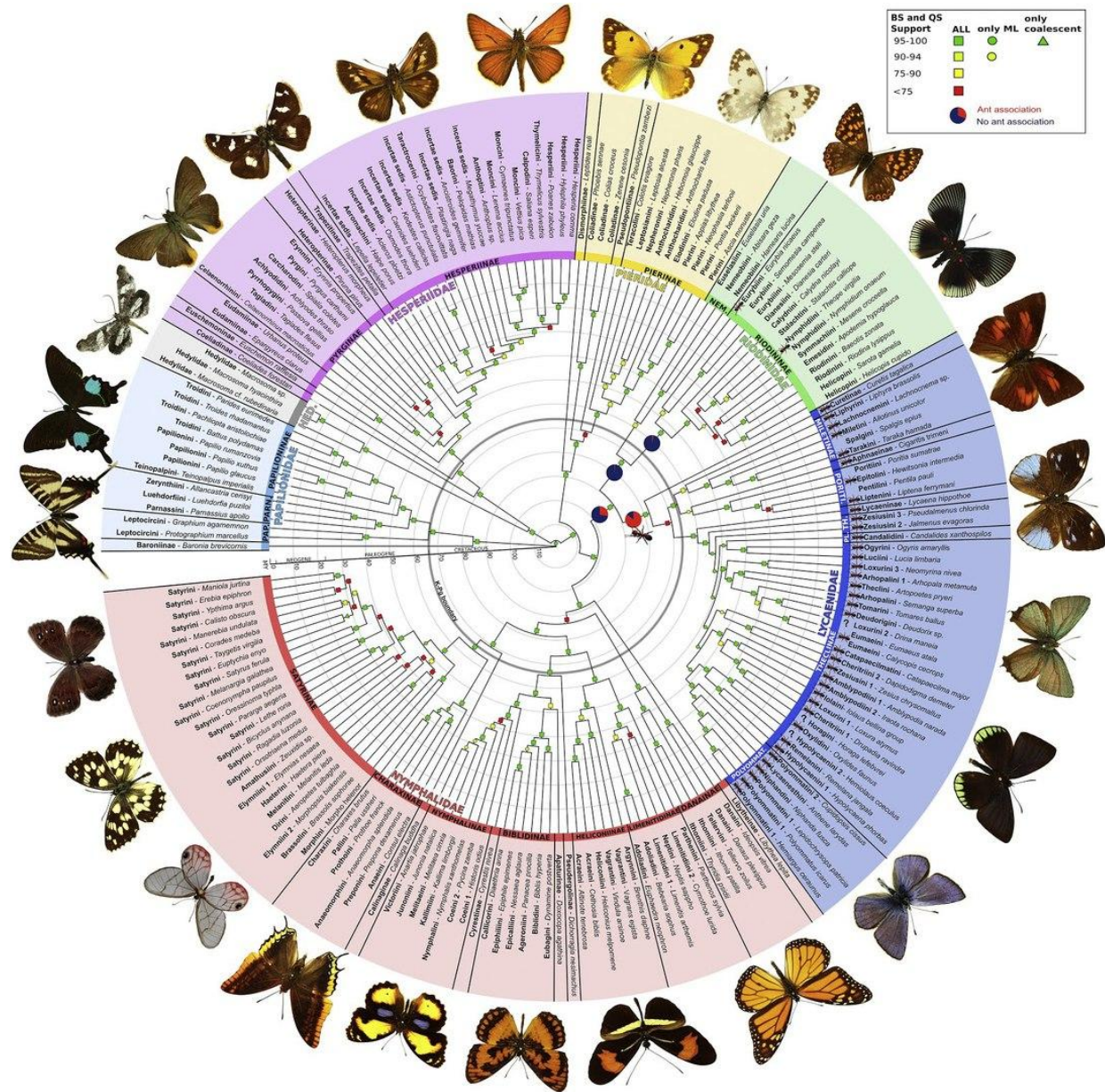
Филогенетические деревья – примеры



Филогенетические деревья – примеры

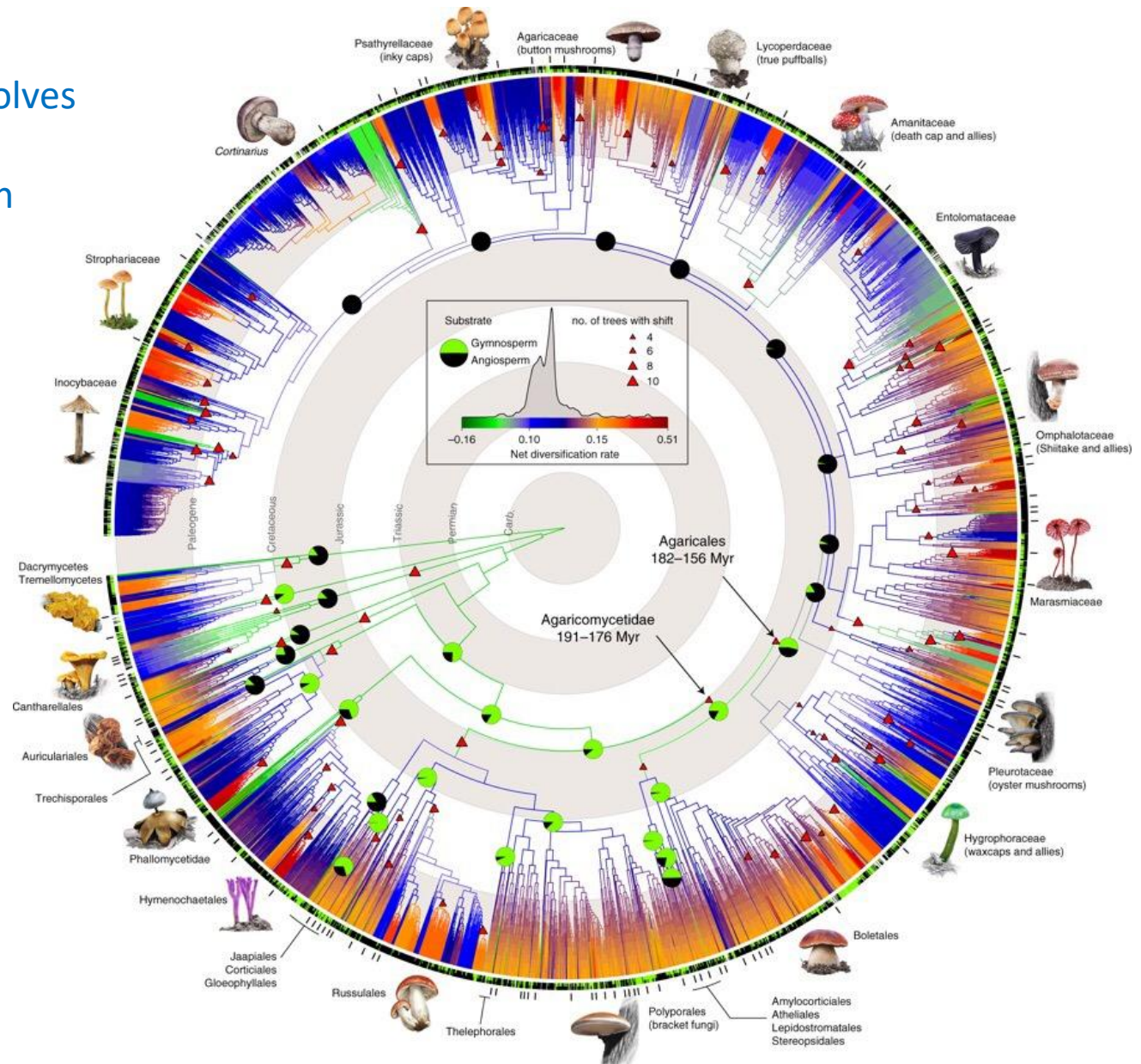


Филогенетические деревья – такие разные



Филогенетические деревья – такие разные

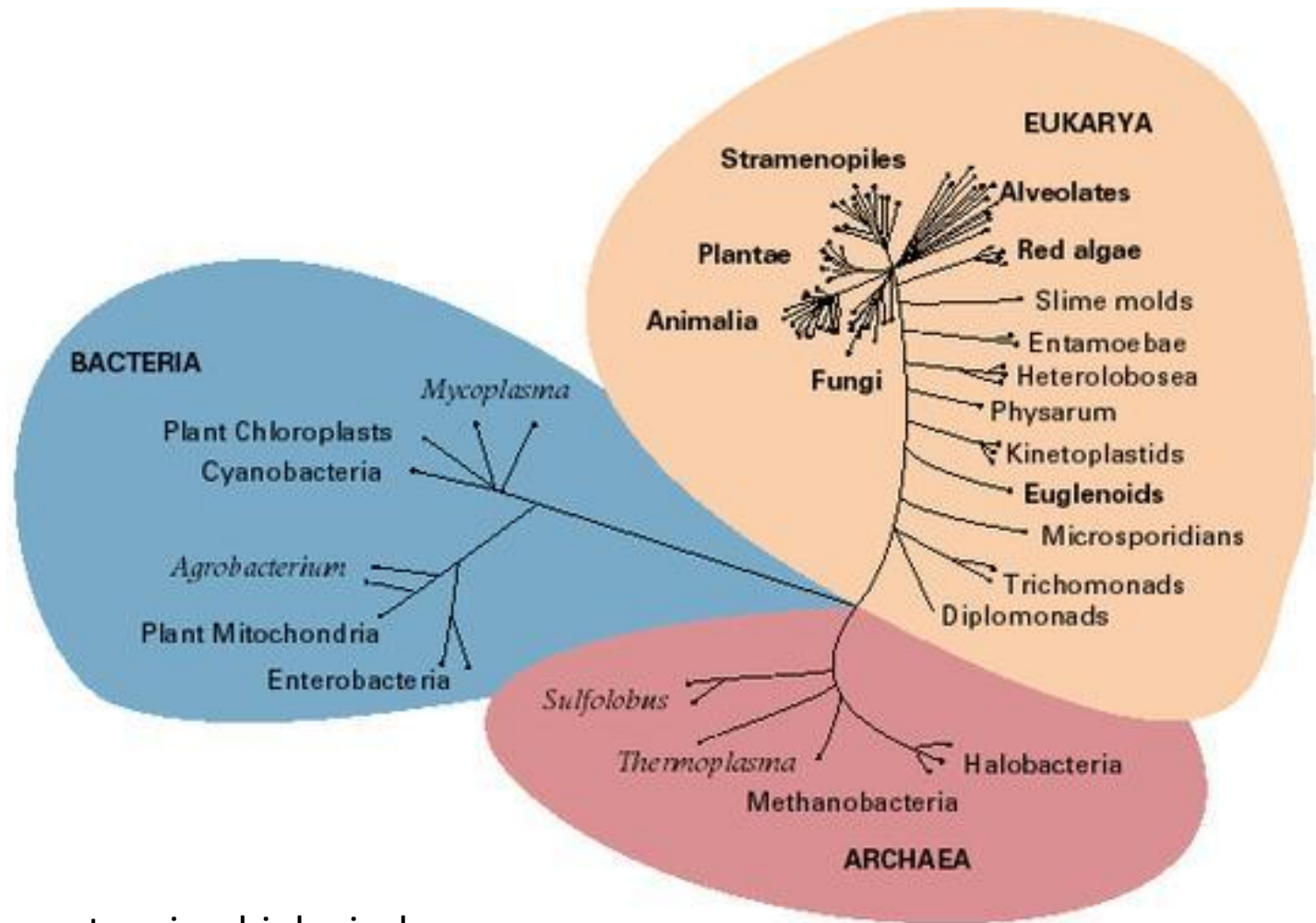
Megaphylogeny resolves
global patterns of
mushroom evolution
(2019)



Phylogenetic trees: so different



Карл Бёзе
(1928 – 2012)



The three-domain system is a biological classification introduced by Carl Woese et al. in 1990

Филогенетические деревья. LUCA

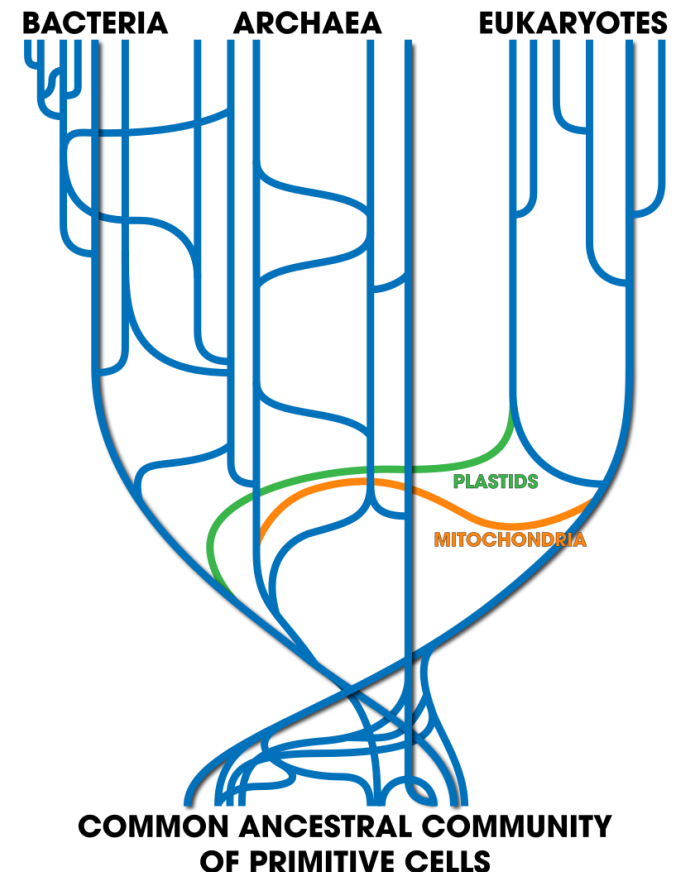
Last Universal Common Ancestor (Darwin, 1859).

Современное представление о существовании такого организма основано на **сопоставлении геномов** (2000 – ...).

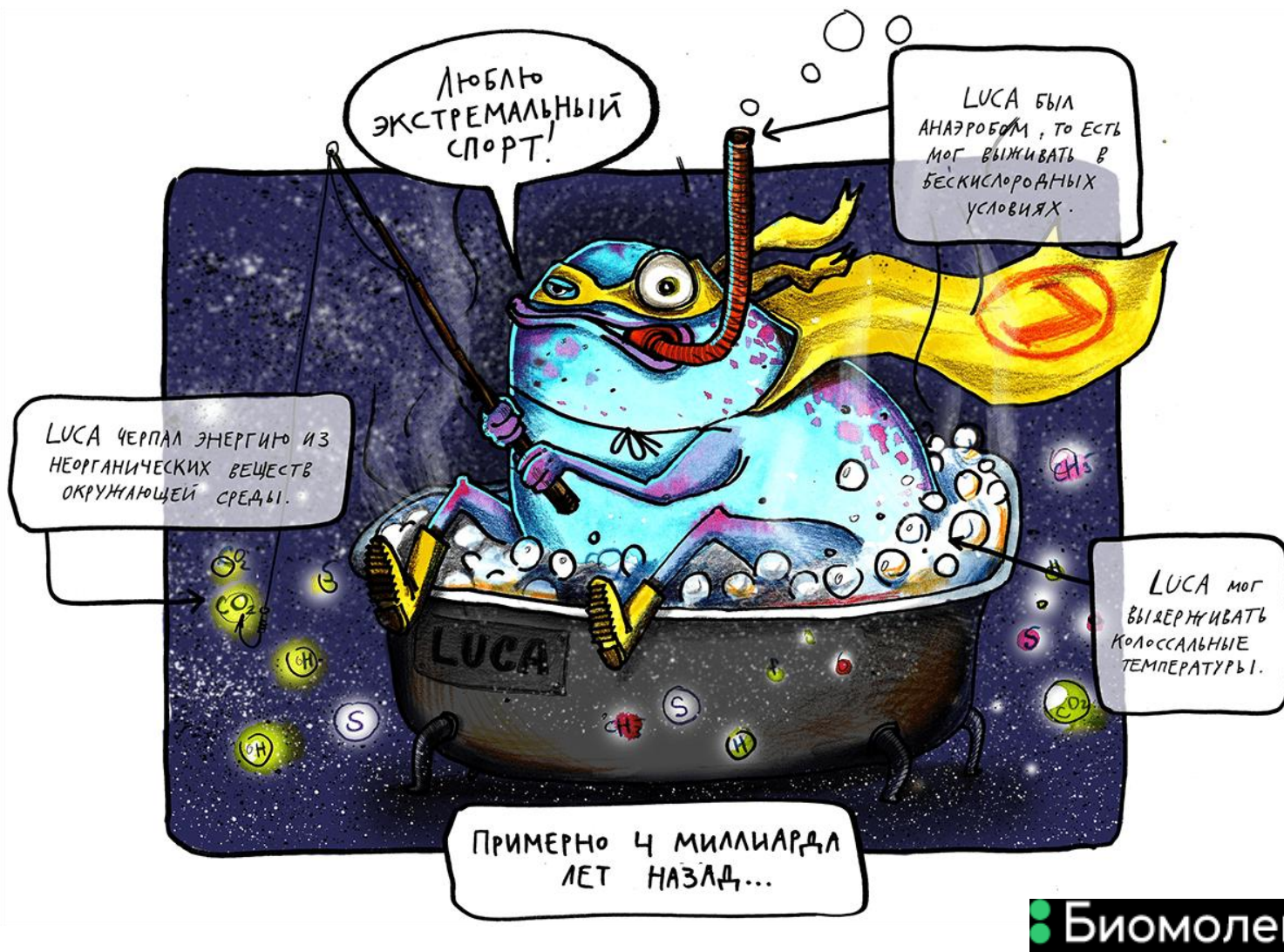
Показано, что наличие общих генов является скорее следствием их общего происхождения, а не горизонтального переноса (Theobald, 2010).

На основании анализа 6,1 млн белок-кодирующих генов можно предположить, что LUCA обладал 355 генами, схожими с генами современных организмов, что даёт возможность детально представить его метаболизм (2016).

Comparative genomics, minimal gene-sets and the last universal common ancestor (Koonin, 2003)



Last Universal Common Ancestor

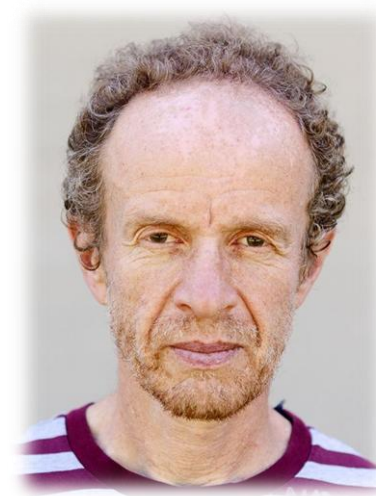
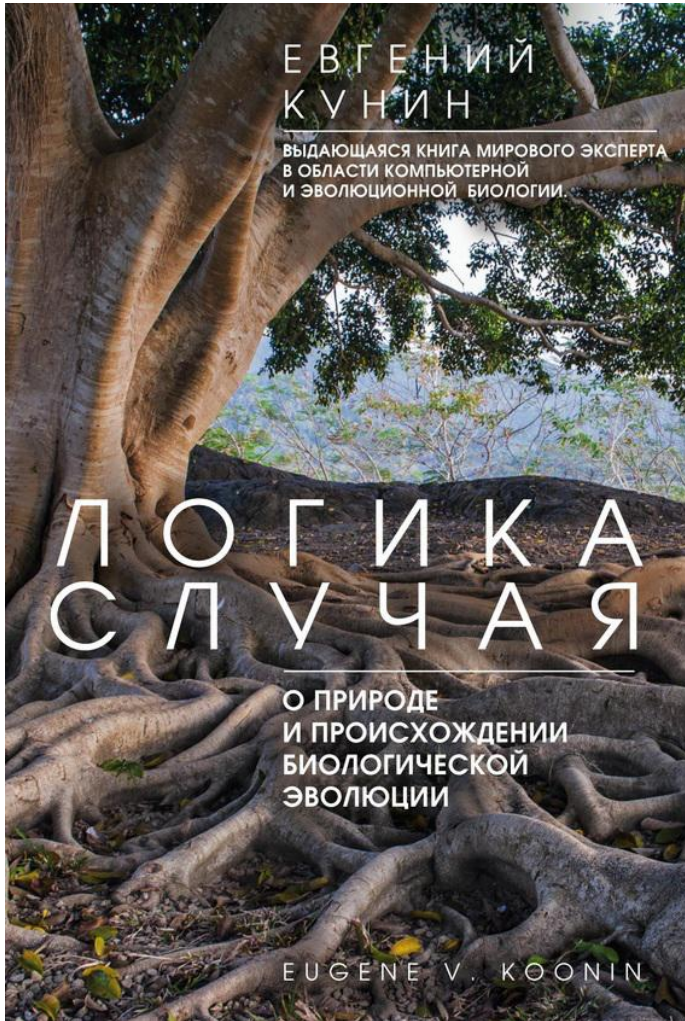


В диких условиях: как жил последний всеобщий предок LUCA.

Что почитать?



Что почитать?



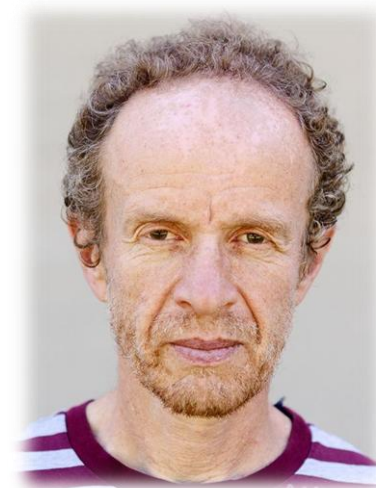
Евгений Кунин
EN = 2

(Биофак МГУ, 1978)

Один из самых цитируемых биологов

Член Национальной академии наук США (2016)

Иностранный член Российской академии наук (2019)



Евгений Кунин
EN = 2

«**В XXI веке** вопрос о необходимости перевода научной литературы с английского на какие-либо другие языки, мягко говоря, неоднозначен. Научные тексты теперь публикуются по-английски, и **умение их читать на этом языке – элементарное требование профессиональной пригодности.**»

Благодарю за внимание!