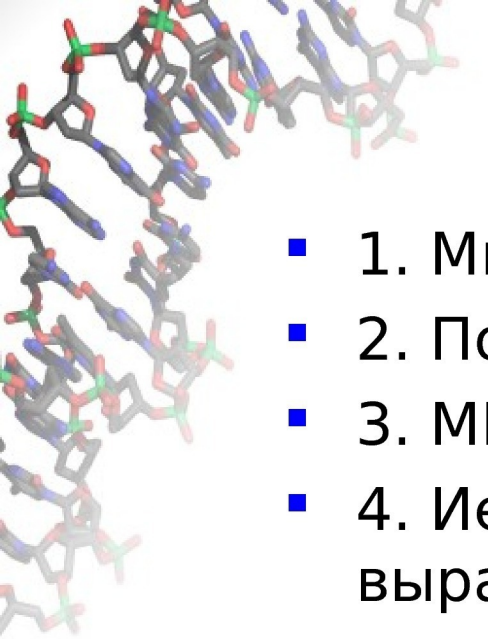
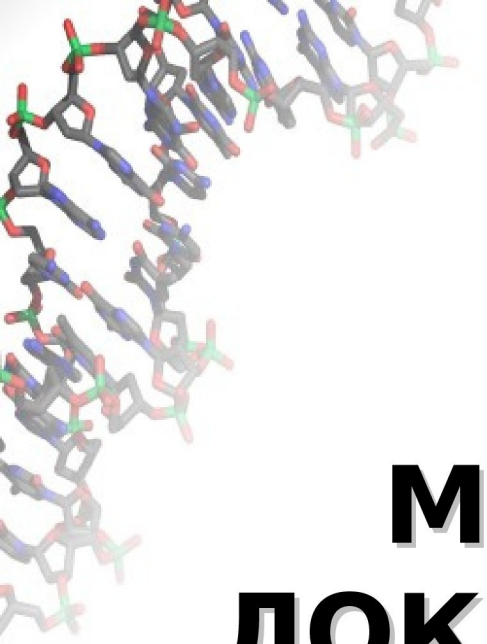


План занятия*

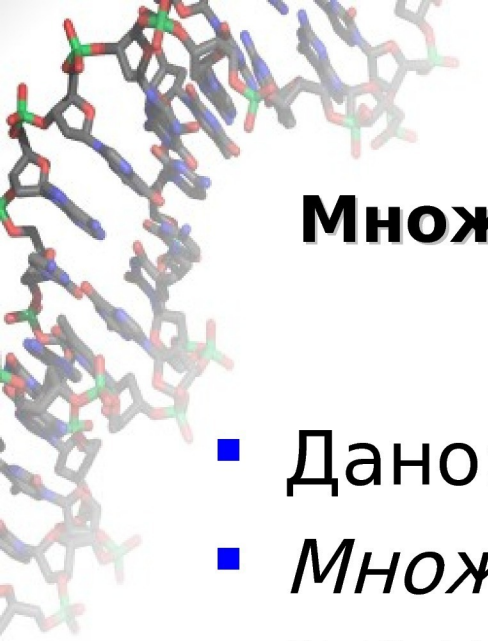
- 1. Множественное выравнивание и HMM
- 2. Поиск локальных множественных сходств.
- 3. MEME и Gibbs sampler
- 4. Иерархическое глобальное множественное выравнивание.



Часть 1

МНОЖЕСТВЕННЫЕ ЛОКАЛЬНЫЕ СХОДСТВА

.



Множественные локальные сходства.

- Дано: тексты *S1, …, Sn*
- *Множественное локальное сходство* – это набор фрагментов этих последовательностей *F1, …, Fk* таких, что фрагменты похожи.

Множественные локальные сходства. Нужны уточнения

- Дано: тексты *S1, …, Sn*
- *Множественное локальное сходство* – это ***набор фрагментов*** этих последовательностей *F1, …, Fk* таких, что фрагменты ***похожи.***

Множественные локальные сходства. Уточнение.

- Дано: *S1, …, Sn*
- *Множественное локальное сходство* – это набор фрагментов этих последовательностей *F1, …, Fk* таких, что фрагменты ***похожи***.
- 1. Количество фрагментов в тексте:

  - ровно 1 в каждом,

  - не менее 1 в каждом,

    - сколько получится (= неточные повторы в объединенном тексте)

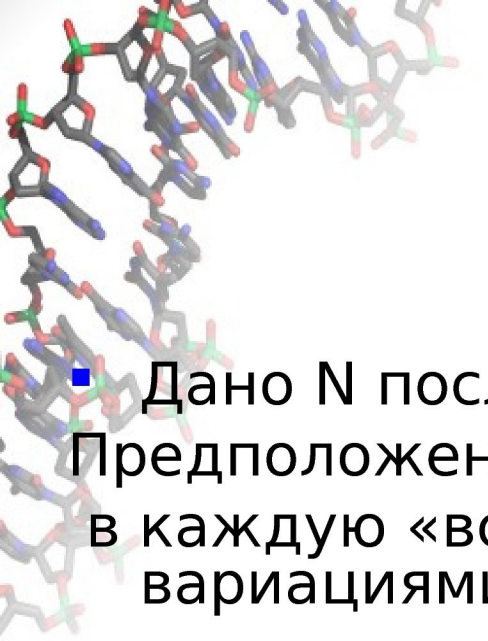
Множественные локальные сходства. Уточнение

- Дано: *S1, …, Sn*
- *Множественное локальное сходство* – это набор фрагментов этих последовательностей *F1, …, Fk* таких, что фрагменты *похожи.*
- 2. Вид сходства:

  - допускаются ли делеции/вставки;

    - порог на качество сходств –

      парный или интегральный



Поиск неизвестного мотива
The Motif Finding Problem

- Дано N последовательностей

Предположение: это случайные последовательности;
в каждую «встроен» мотив – данное слово с вариациями (без делеций)

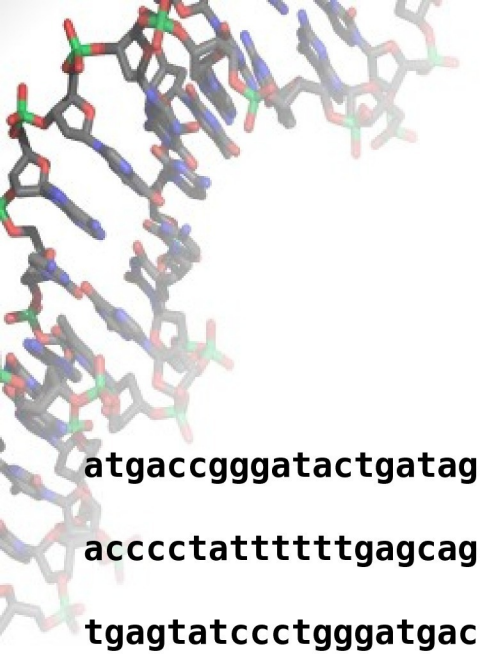
```
cctgatagacgctatctggctatccacgtacgtaggtcctctgtgcgaatctatgcgtttccaaccat

agtactggtgtacatttgatacgtacgtacaccggcaacctgaaacaaacgctcagaaccagaagtgc

aaacgtacgtgcaccctctttcttcgtggctctggccaacgagggctgatgtataagacgaaaatttt

agcctccgatgtaagtcatagctgtaactattacctgccacccctattacatcttacgtacgtataca

ctgttatacaacgcgtcatggcggggtatgcgttttggtcgtcgtacgctcgatcgttaacgtacgtc
```



Пример. Где мотив? ☺

atgaccgggatactgatagaagaaaggttgggggcgtacacattagataaacgtatgaagtacgttagactcggcgccgccg

acccctatttttgagcagatttagtgacctggaaaaaaaatttgagtacaaaactttccgaatacaataaaacggcggga

tgagtatccctgggatgacttaaaataatggagtggtgctctcccgatttttgaatatgtaggatcattcgccagggtccga

gctgagaattggatgcaaaaaaagggattgtccacgcaatcgcgaaccaacgcggacccaaaggcaagaccgataaaggaga

tccctttgcggtaatgtgccgggaggctggttacgtagggaagccctaacggacttaatataataaaggaagggcttatag

gtcaatcatgttcttgtgaatggatttaacaataagggctgggaccgcttggcgcacccaaattcagtgtgggcgagcgcaa

cggttttggcccttgttagaggcccccgtataaacaaggagggccaattatgagagagctaatctatcgcgtgcgtgttcat

aacttgagttaaaaaatagggagccctggggcacatacaagaggagtcttccttatcagttaatgctgtatgacactatgta

ttggcccattggctaaaagcccaacttgacaaatggaagatagaatccttgcatactaaaaaggagcggaccgaaagggaag

ctggtgagcaacgacagattcttacgtgcattagctcgcttccggggatctaatagcacgaagcttactaaaaaggagcgga

Ответ: AAAAAAAAGGGGGGG (длина 15) с 4 заменами (15, 4)- мотив

atgaccgggatactgatAgAAgAAAGGttGGGggcgtacacattagataaacgtatgaagtacgttagactcggcgccgccg

acccctatttttgagcagatttagtgacctggaaaaaaatttgagtacaaaacttttccgaatacAAtAAAAcGGcGGGa

tgagtatccctgggatgacttAAAAtAAtGGaGtGGtgctctcccgattttgaatatgtaggatcattcgccagggtccga

gctgagaattggatgcAAAAAAAGGGattGtccacgcaatcgcgaaccaacgcggacccaaaggcaagaccgataaaggaga

tccctttgcggtaatgtgccgggaggctggttacgtagggaagccctaacggacttaatAtAAtAAAGGaaGGGcttatag

gtcaatcatgttcttgtgaatggatttAAcAAtAAGGGctGGgaccgcttggcgcacccaaattcagtgtgggcgagcgcaa

cggttttggcccttgttagaggcccccgtAtAAAcAAGGaGGGccaattatgagagagctaatctatcgcgtgcgtgttcat

aacttgagttAAAAAAtAGGGaGccctggggcacatacaagaggagtcttccttatcagttaatgctgtatgacactatgta

ttggcccattggctaaaagcccaacttgacaaatggaagatagaatccttgcatActAAAAAGGaGcGGaccgaaagggaag

ctggtgagcaacgacagattcttacgtgcattagctcgcttccggggatctaatagcacgaagcttActAAAAAGGaGcGGa

Сложность; 2 экземпляра отличаются в 8 > 15/2 позициях

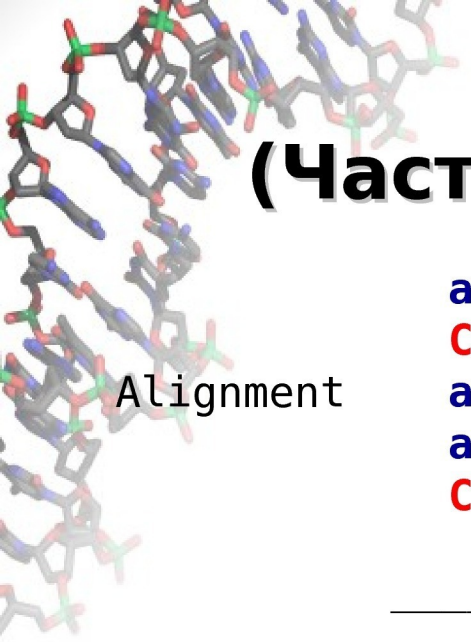
Определения

- Будем задавать мотив вектором начальных позиций его вхождений в каждую из последовательностей

$$\boldsymbol{s} = (s_1, s_2, s_3, \ldots, s_t)$$

Как оценить качество мотива?

Подготовка:
(Частотный) Профиль и консенсус



Alignment

```
a G g t a c T t
C c A t a c g t
a c g t T A g t
a c g t C c A t
C c g t a c g G
```

Profile

```
A  3 0 1 0 3 1 1 0
C  2 4 0 0 1 4 0 0
G  0 1 4 0 0 0 3 1
T  0 0 0 5 1 0 1 4
```

Consensus

```
A C G T A C G T
```

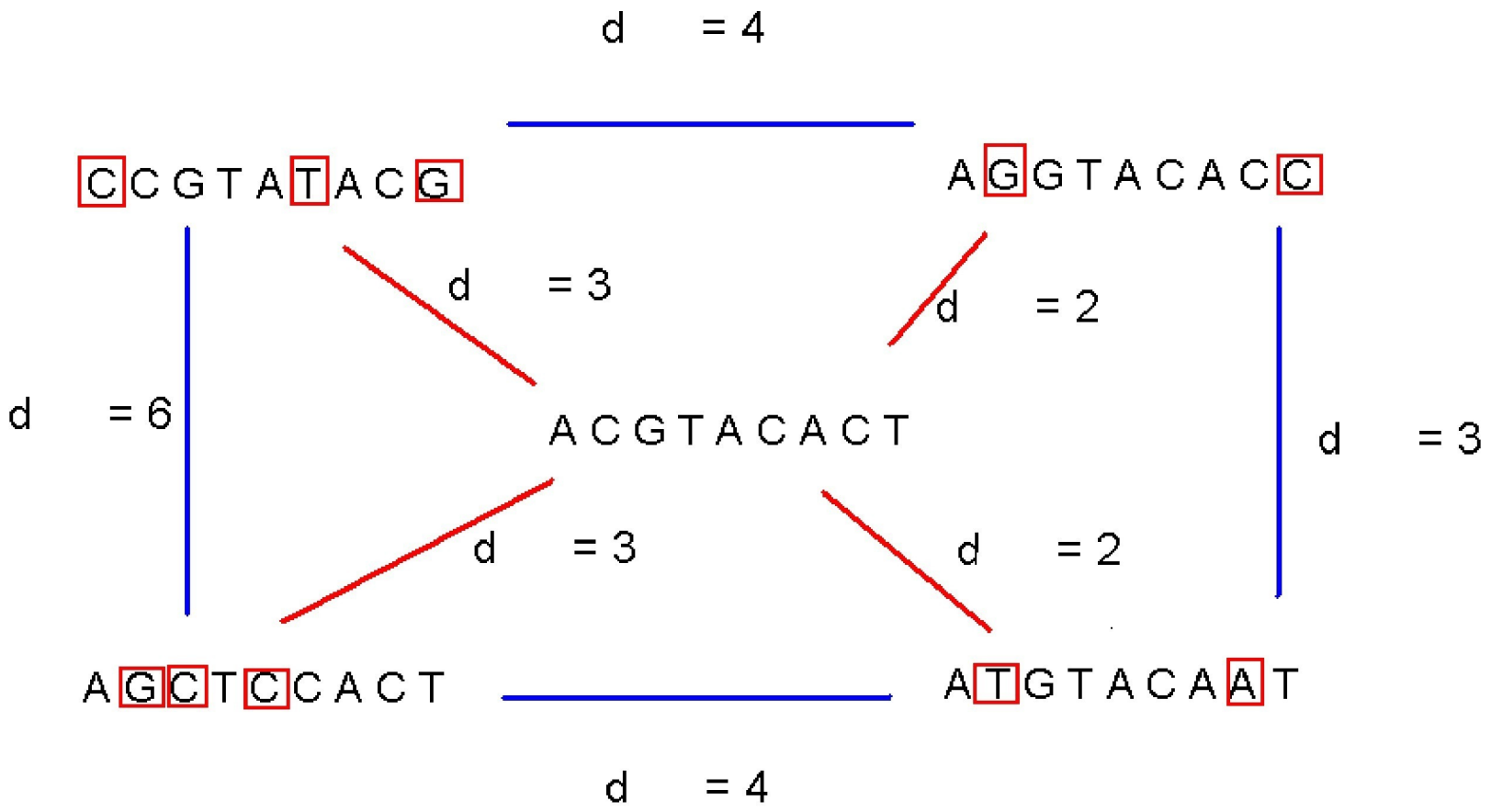
- Расположим вхождения друг над другом
  $\mathbf{s} = (s_1, s_2, \ldots, s_t)$
- Строим матрицу частот (частотный профиль)
- Консенсус-нуклеотид – наиболее частый нуклеотид в своей колонке

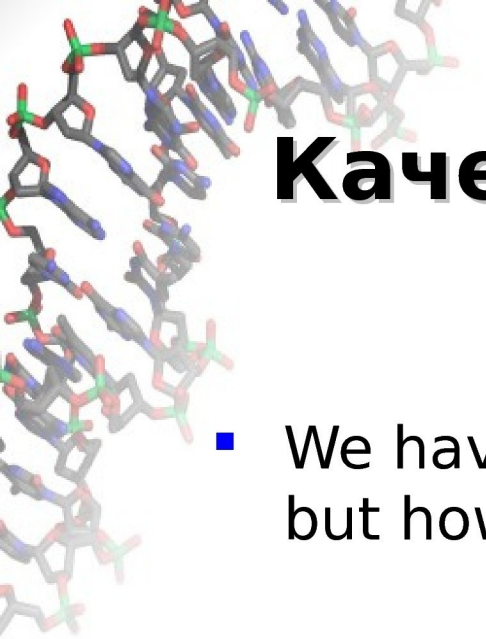
Консенсус

- Можно рассматривать консенсус, как предковый мотив

- Расстояние от консенсуса до каждого из вхождений *как правило* МЕНЬШЕ, чем расстояние между наиболее непохожими вхождениями мотива

Пример


d = 4
CCGTATACG
AGGTACACC
d = 3
d = 2
d = 6
ACGTACACT
d = 3
d = 3
d = 2
AGCTCCACT
ATGTACAAT
d = 4

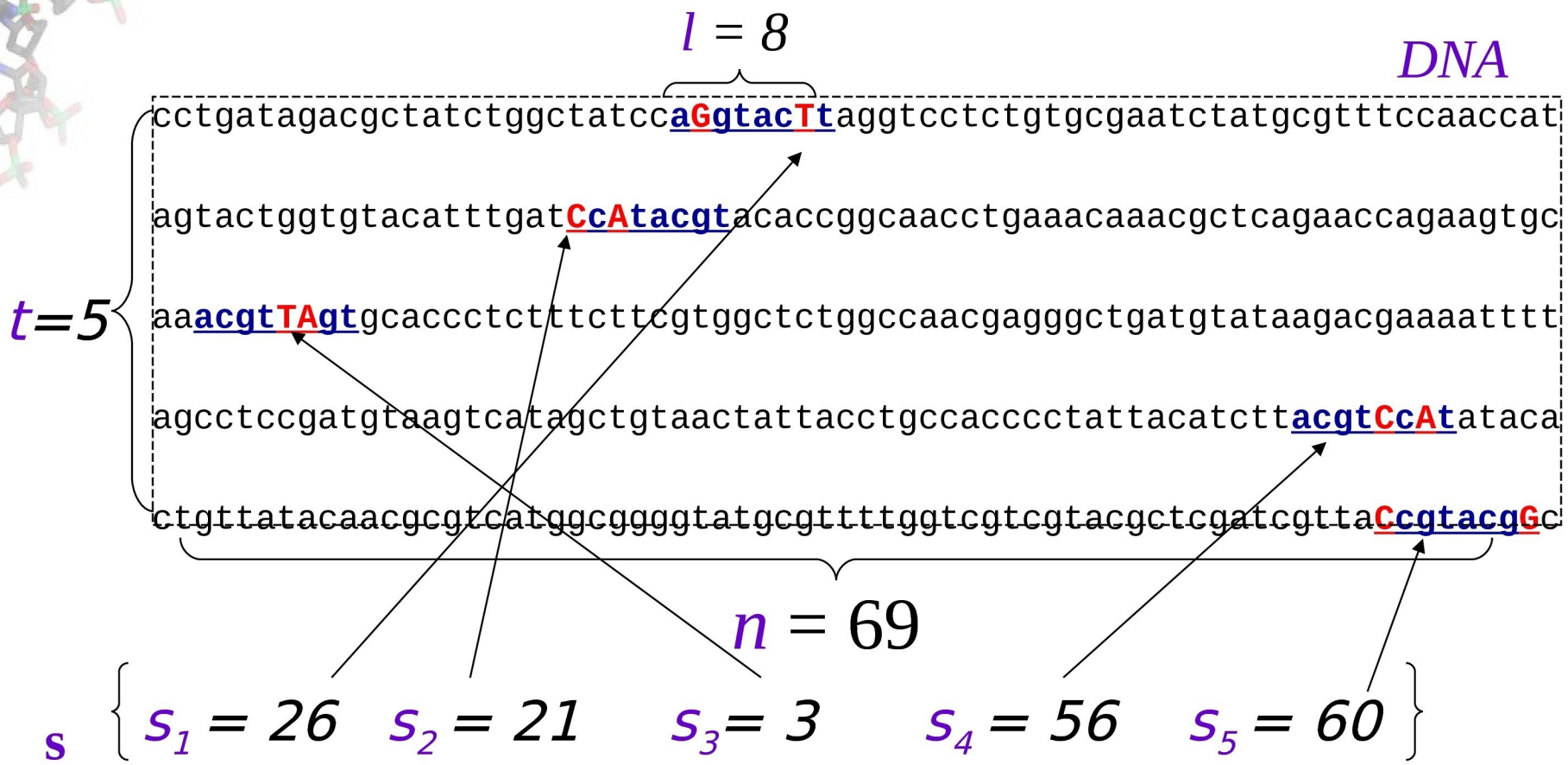


Качество мотива = качество консенсуса

- We have a guess about the consensus sequence, but how “good” is this consensus?

- Need to introduce a scoring function to compare different guesses and choose the “best” one.

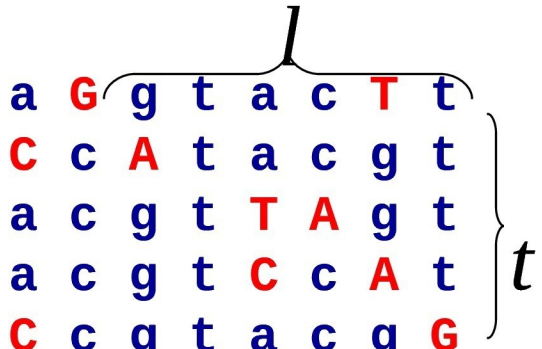
Пример


l = 8
DNA
cctgatagacgctatctggctatccaGgtacTtaggtcctctgtgcgaatctatgcgtttccaaccat
agtactggtgtacatttgatCcAtacgtacaccggcaacctgaaacaaacgctcagaaccagaagtgc
aaacgtTAgtgcaccctctttcttcgtggctctggccaacgagggctgatgtataagacgaaaatttt
agcctccgatgtaagtcatagctgtaactattacctgccacccctattacatcttacgtCcAtataca
ctgttatacaacgcgtcatggcggggtatgcgttttggtcgtcgtacgctcgatcgttaCcgtacgGc
t=5
n = 69
s
s1 = 26
s2 = 21
s3 = 3
s4 = 56
s5 = 60

Определение веса мотива (при заданном наборе из *t* последовательностей)

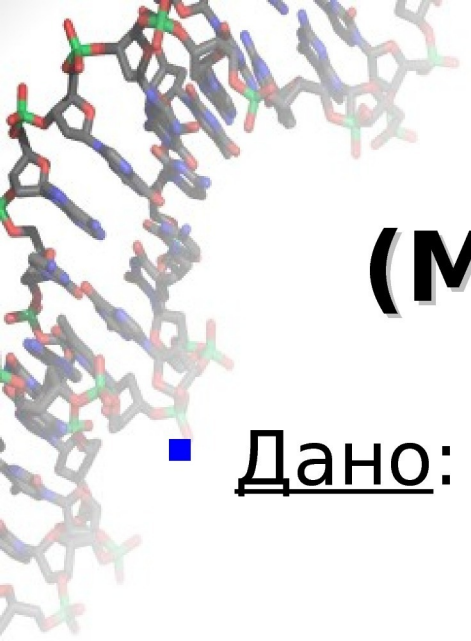
- *Дано:* **s** $= (s_1, \ldots s_t)$:

$$Score(\mathbf{s}) = \sum_{i=1}^{l} \max_{k \in \{A,T,C,G\}} count(k,i)$$



A	3	0	1	0	3	1	1	0
C	2	4	0	0	1	4	0	0
G	0	1	4	0	0	0	3	1
T	0	0	0	5	1	0	1	4
Consensus	a	c	g	t	a	c	g	t

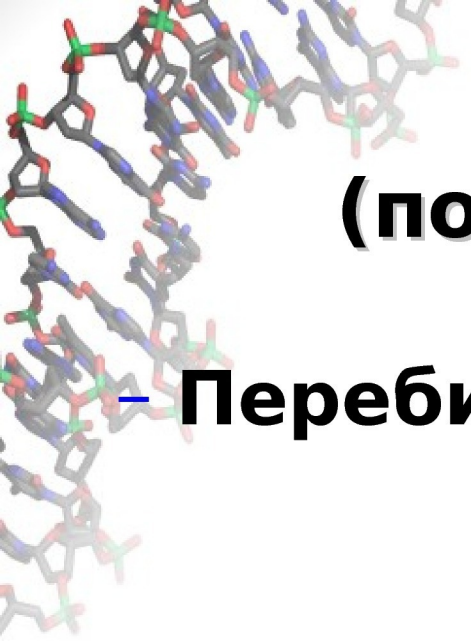
Score 3+4+4+5+3+4+3+4 =30



Постановка задачи (Motif Finding Problem)

- Дано: ***t*** последовательностей длины ***n;***
  длина мотива ***L***

- Найти: Вектор стартовых позиций **s** = (s_1, s_2, … s_t), для которого *Score*(**s**) будет максимальным



Решение 1 (полный перебор по данным)

- Перебираем все возможные вектора ***s***

$$s_i = [1, \ldots, n-l+1]$$

$$i = [1, \ldots, t]$$

Решение 2 (полный перебор по мотивам)

- Просматриваем все возможные слова длины *L*
- Для каждого слова *w* длины *L*

  -- находим его наилучшее вхождение в каждой строке;

  -- оцениваем полученный профиль

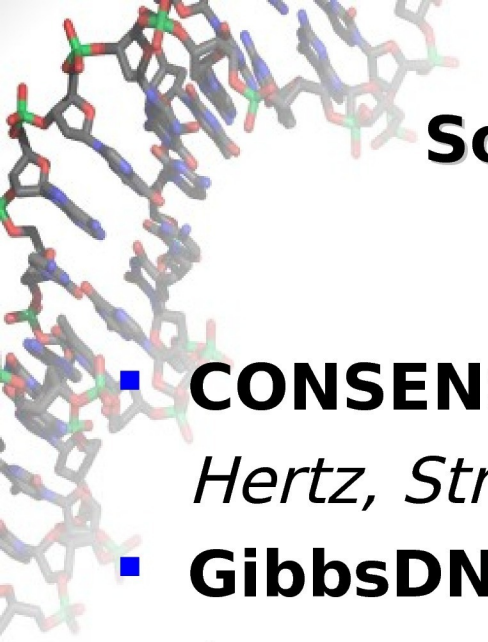
CONSENSUS (*Hertz, Stormo (1989)*): Greedy Motif Search

- Find two closest l-mers in sequences 1 and 2 and forms
  2 x l alignment matrix with *Score(**s**,2)*

- At each of the following *t-2* iterations CONSENSUS finds a "best" *l*-mer in sequence *i* from the perspective of the already constructed *(i-1) x l* alignment matrix for the first *(i-1)* sequences

- In other words, it finds an *l*-mer in sequence *i* maximizing
  *Score(**s**,i)*
  under the assumption that the first *(i-1) l*-mers have been already chosen

- CONSENSUS sacrifices optimal solution for speed: in fact the bulk of the time is actually spent locating the first 2 *l*-mers



Some Motif Finding Programs

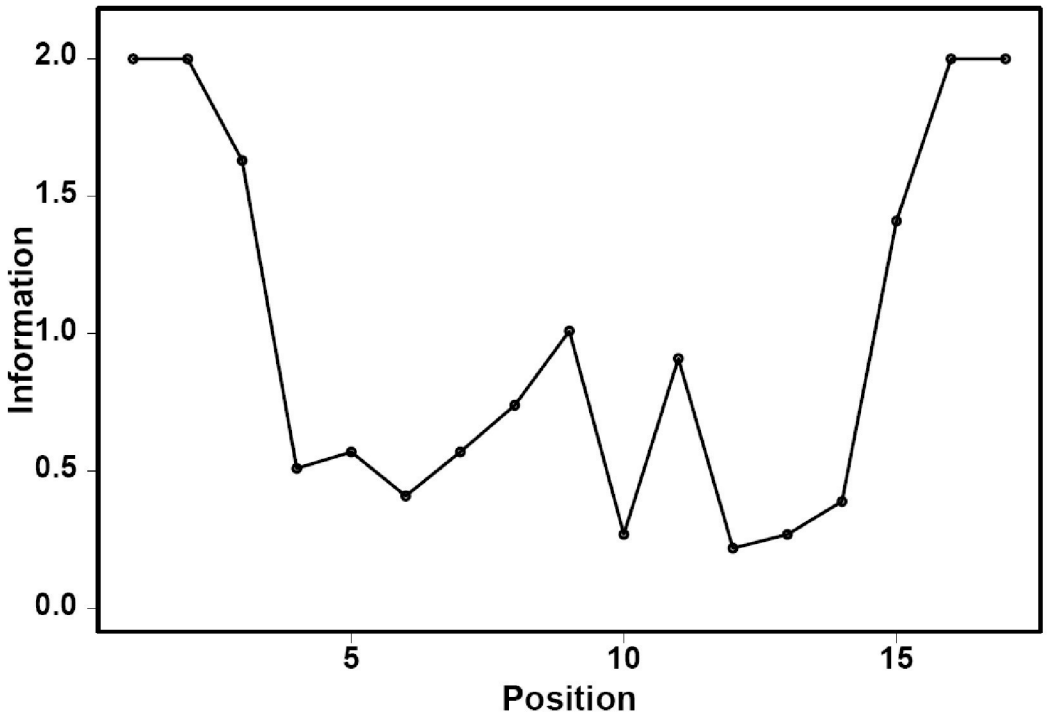
- **CONSENSUS**
  Hertz, Stromo (1989)
- **GibbsDNA**
  Lawrence et al (1993)
- **MEME**
  *Bailey, Elkan (*1995)
- **RandomProjections**
  Buhler, Tompa (2002)
- **MULTIPROFILER** *Keich, Pevzner (2002)*
- **MITRA**
  Eskin, Pevzner (2002)
- **Pattern Branching**
  Price, Pevzner (2003)

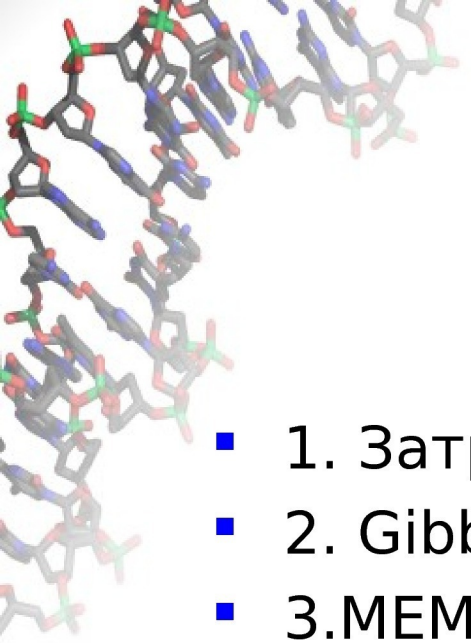
Gal4 Motif Information Content

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
A	0	0	0	$\frac{7}{14}$	$\frac{1}{14}$	$\frac{1}{14}$	$\frac{9}{14}$	0	$\frac{6}{14}$	$\frac{1}{14}$	0	$\frac{3}{14}$	$\frac{1}{14}$	$\frac{2}{14}$	0	0	0
C	$\frac{14}{14}$	0	$\frac{1}{14}$	$\frac{3}{14}$	$\frac{3}{14}$	$\frac{6}{14}$	$\frac{3}{14}$	$\frac{8}{14}$	0	$\frac{5}{14}$	$\frac{3}{14}$	$\frac{7}{14}$	$\frac{5}{14}$	$\frac{3}{14}$	$\frac{12}{14}$	$\frac{14}{14}$	0
G	0	$\frac{14}{14}$	$\frac{13}{14}$	$\frac{4}{14}$	$\frac{9}{14}$	$\frac{6}{14}$	$\frac{1}{14}$	$\frac{5}{14}$	0	$\frac{6}{14}$	$\frac{1}{14}$	$\frac{2}{14}$	$\frac{6}{14}$	$\frac{1}{14}$	$\frac{2}{14}$	0	$\frac{14}{14}$
T	0	0	0	0	$\frac{1}{14}$	$\frac{1}{14}$	$\frac{1}{14}$	$\frac{1}{14}$	$\frac{8}{14}$	$\frac{2}{14}$	$\frac{10}{14}$	$\frac{2}{14}$	$\frac{2}{14}$	$\frac{8}{14}$	0	0	0

Information =

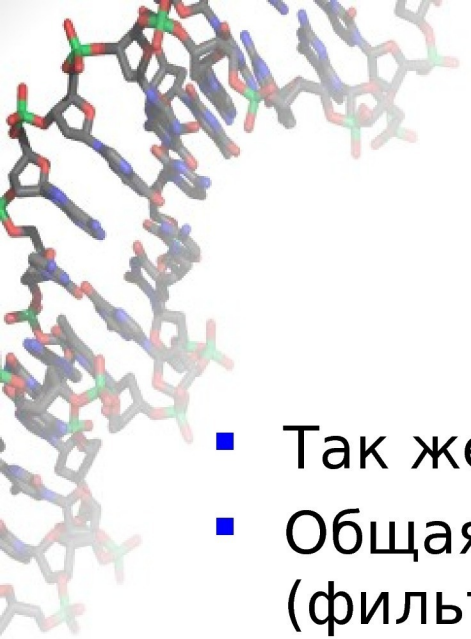
$$2 + \sum_{b=\{A,C,G,T\}} p(b)\log_2 p(b)$$





Как искать МЛС

- 1. Затравки
- 2. Gibbs Sampler (случай 1 из каждого)
- 3.MEME



Затравки

- Так же, как и в парном случае.
- Общая идея: сокращение пространства поиска (фильтрация).
- Разреженные затравки и пр.

Gibbs Motif Sampling

- Gibbs Motif Sampler
  - Bayesian model, prior distribution
- Algorithm (MCMC)

Initialization: Randomly select motif start-positions in each sequence

Iterations:

Remove randomly selected sequence k'

- Update frequency matrix
- Randomly select a motif start-position j for k' proportional to:

$$\frac{\text{Probability under motif model}}{\text{Probability under background model}} = \prod_i \frac{p_i(b^k_{j+i-1})}{p_0(b^k_{j+i-1})}$$

Gibbs Motif Sampler http://bayesweb.wadsworth.org/gibbs/gibbs.html

The Gibbs Motif Sampler

(for DNA)

Show advanced options How to enter data?

Email Address:

Please enter the data sequence: (FASTA format) *

Browse...

Prokaryotic Defaults | Prokaryotic Defaults | **Eukaryotic Defaults** | Eukaryotic Defaults

Sampler Mode: ○ Site Sampler ○ Motif Sampler ◉ Recursive Sampler

No. of different motifs (patterns):

Max sites per seq: (recursive sampler)

Motif Width(s):*

Est. total sites for each motif type:

Submit Clear

Expectation-Maximization (EM) Algorithm

- MEME
  - Missing data problem: Expectation-Maximization (EM) Algorithm to obtain maximum likelihood estimates
- EM Algorithm

  Initialization: Set frequency matrix p and p_0

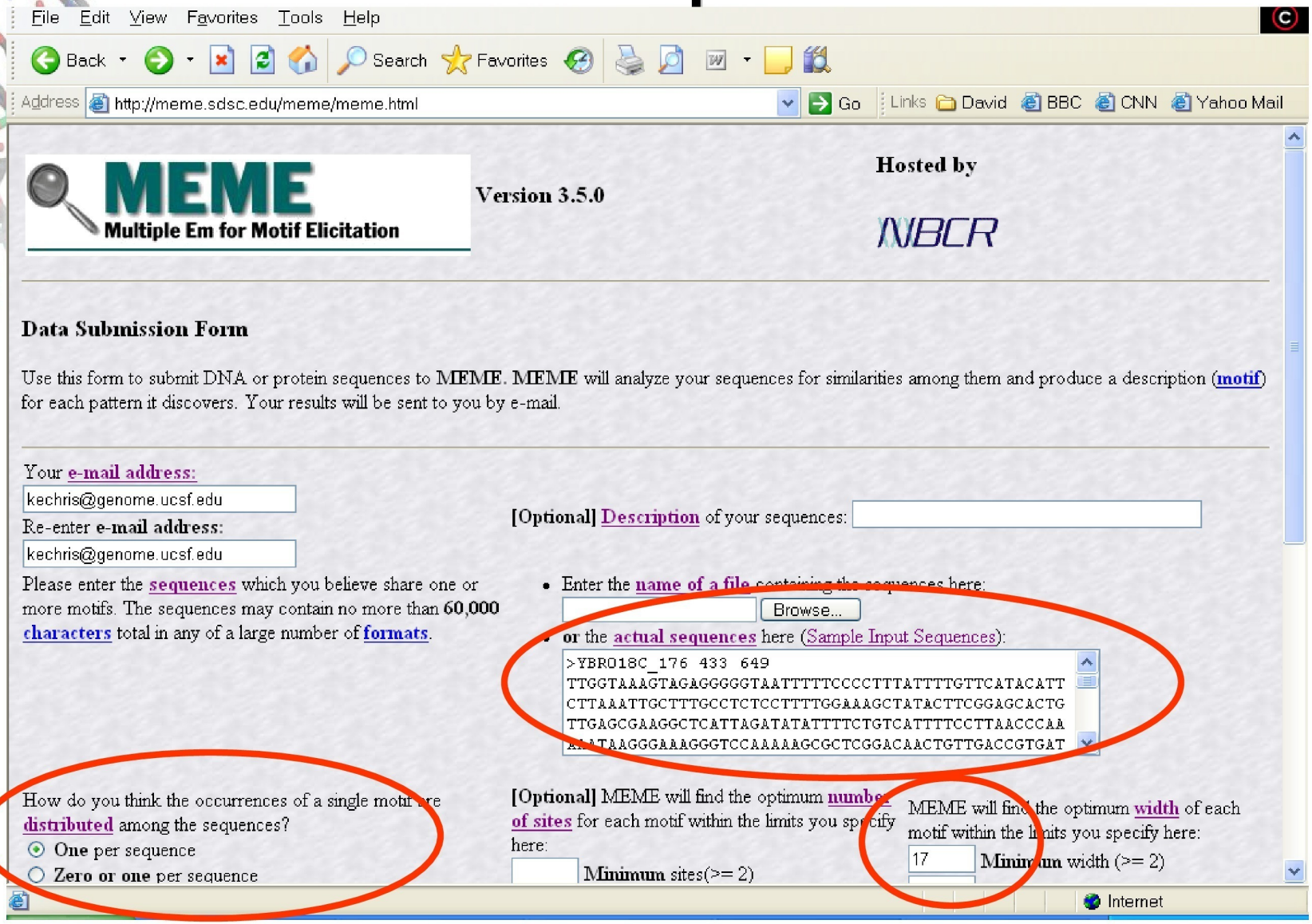
  Iterations:
  - E-step: Calculate probability of motif start-positions

    For each sequence k and position j

    $W_{kj} = Pr(\text{motif start-position} = j \mid p)$
  - M-step: Update frequency matrix estimate

$$\hat{p}_i(b) = \frac{\sum_k \sum_j W_{kj}\, 1(\text{sequence } k, \text{position } j+i-1=b)}{N}, \quad b = \mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}$$

MEME
http://meme.sdsc.edu/meme/website/meme.html

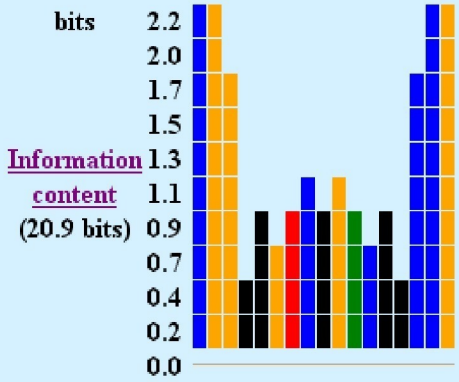


MEME Output

PN **MOTIF 1** **width = 17** **sites = 14** **llr = 203** **E-value = 2.7e-025**

Simplified pos.-specific probability matrix

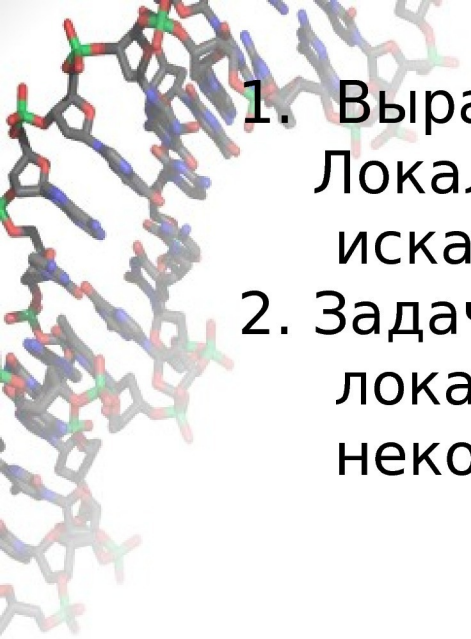
```
A : : : 5: 1615: : 1: 1: : :
C a: 1152: 7: 246549a:
G : a945642: 7: 2511: a
T : : : 1: 1: : 5161: 5: : :
```



Multilevel consensus sequence

```
CGGAGGACTGTCCTCCG
 GCCG A CGGC
```

NAME	START	P-VALUE	SITES	
YBR020W_74	93	1.11e-09	AGCCGCCGAG	CGGGCGACAGCCCTCCG ACGGAAGACT
YBR020W_153	153	1.11e-09	AGCCGCCGAG	CGGGCGACAGCCCTCCG ACGGAAGACT
YBR020W_154	136	1.11e-09	AGCCGCCGAG	CGGGCGACAGCCCTCCG ACGGAAGACT
YBR019C_94	176	1.44e-09	AGTCTTCCGT	CGGAGGGCTGTCGCCCG CTCGGCGGCT



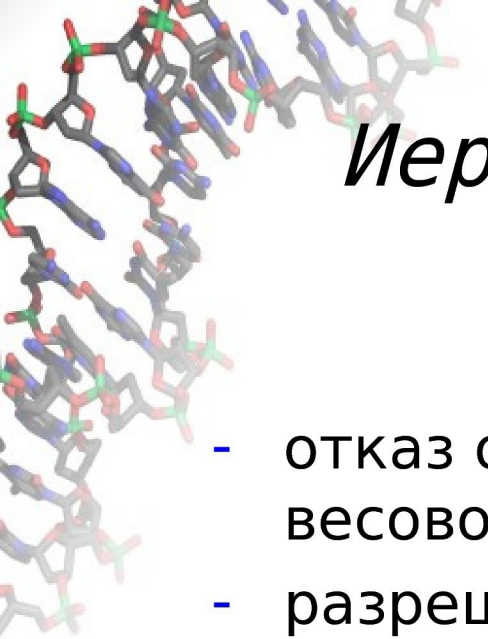
1. Выравнивание – это цепочка локальных сходств. Локальные (неточные) сходства можно легко искать
2. Задача выравнивания – выделить в множестве локальных сходств «разумную» цепочку неконфликтующих локальных сходств

ПРОБЛЕМА:

Чтобы получить детальное выравнивание, исходных локальных сходств должно быть много

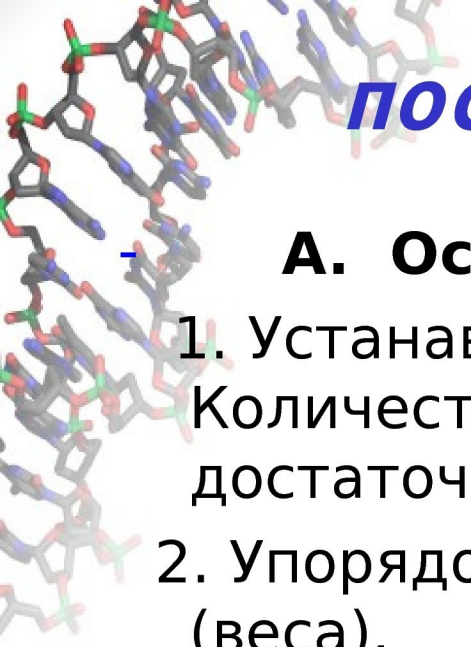
РЕШЕНИЕ:

Иерархическое выравнивание



Иерархическое выравнивание

- отказ от глобальной оптимизации какой-либо весовой весовой функции;
- разрешение конфликта между сходствами производится локально
- иерархическая процедура



Иерархическая процедура построения цепочки локальных сходств

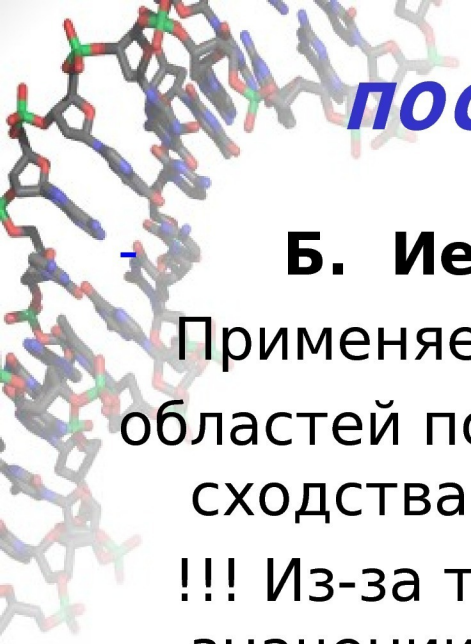
- **А. Основной шаг** (жадный алгоритм)

1. Устанавливаем порог статистической значимости. Количество локальных сходств –кандидатов достаточно мало.
2. Упорядочиваем сходства по убыванию значимости (веса).
3. Берем очередное сходство. Если оно ни с кем не конфликтует – включаем в цепочку.

   Иначе – отбрасываем [на самом деле – откладываем для дальнейшего изучения]

** Техническая проблема: повторы (“low complexity”).

Нужно отдельно фильтровать.



Иерархическая процедура построения цепочки локальных сходств

- **Б. Иерархический переход**

Применяем основной шаг отдельно к каждой из областей поиска между отобранными локальными сходствами.

!!! Из-за того, что области стали меньше, при том же значении порога

сходства становятся значимыми при меньшем весе

Поэтому промежутки между сходствами в цепочке будут постепенно заполняться

Иерархическая процедура построения цепочки локальных сходств

- **А. Основной шаг**

Строим множество статистически значимых сходств и жадным алгоритмом выбираем «остовную цепочку».

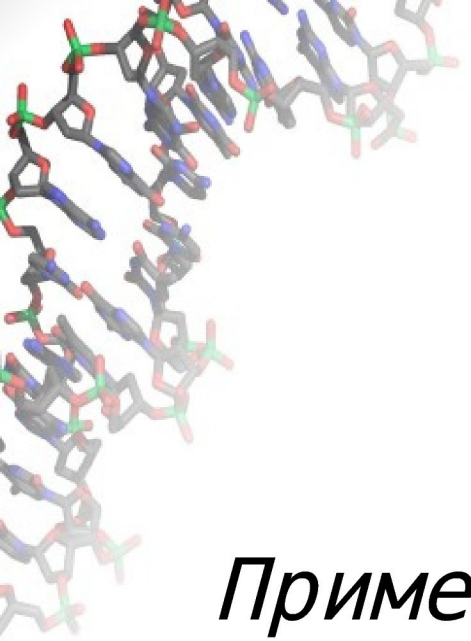
Б. Иерархический переход

Применяем основной шаг отдельно к каждой из областей поиска между отобранными локальными сходствами.

В. Итерирование

Повторяем шаг **Б** до тех пор, пока появляются новые значимые сходства.

Время определяется принятым уровнем значимости (а не минимальным допустимым весом сходства)



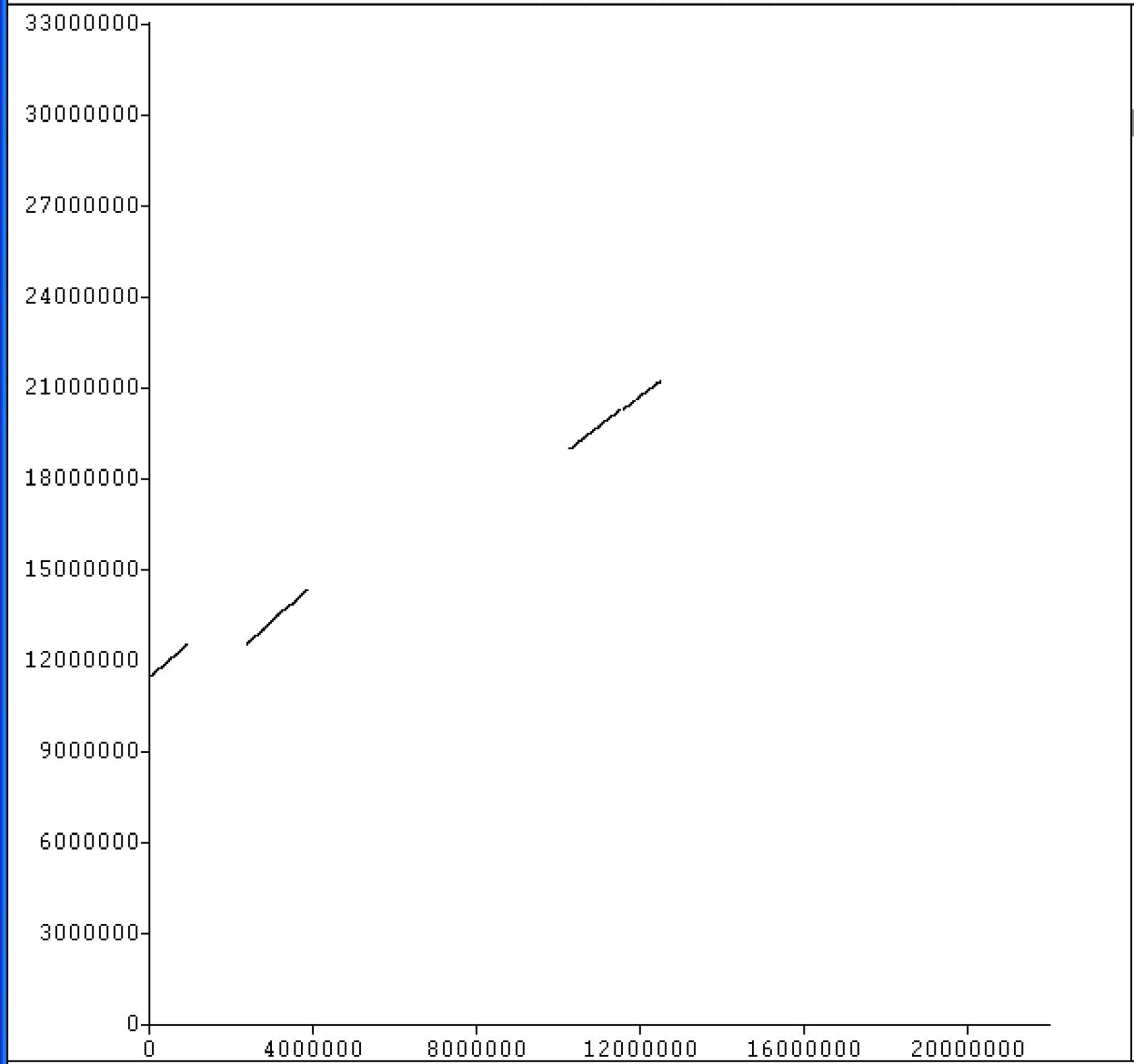
Пример: иерархическое геномное выравнивание (длина ~ 10^7)

Owen v1.4

File Construct Conflict Filter Select Edit Window Help

1 .. 20206255 D.Viril scaffold_12963 type=golden_path_region; loc=scaffold_12963:1..20206255; ID=scaffold_12963; dbxref=GB:CH94064

1 .. 32352404 D.Mojaven scaffold_6500 type=golden_path_region; loc=scaffold_6500:1..32352404; ID=scaffold_6500; dbxref=GB:CH9338



88% 136722 .. 136877 (156) 11585441 .. 11585593 (153

86% 136946 .. 137141 (196) 11585773 .. 11585979 (207

78% 137166 .. 137314 (149) 11586093 .. 11586230 (138

85% 137789 .. 137898 (110) 11587018 .. 11587135 (118

84% 137988 .. 138183 (196) 11587256 .. 11587452 (197

86% 138242 .. 138344 (103) 11587538 .. 11587645 (108

83% 138410 .. 138557 (148) 11587753 .. 11587903 (151

84% 138825 .. 139049 (225) 11588378 .. 11588594 (217

89% 139775 .. 140039 (265) 11589342 .. 11589603 (262

81% 140195 .. 140463 (269) 11589698 .. 11589984 (287

85% 140553 .. 140640 (88) 11590135 .. 11590222 (88)

85% 141082 .. 141235 (154) 11590839 .. 11590995 (157

95% 141304 .. 141365 (62) 11591047 .. 11591109 (63)

77% 141390 .. 141484 (95) 11591139 .. 11591239 (101)

79% 141529 .. 141733 (205) 11591256 .. 11591474 (219

82% 142350 .. 142536 (187) 11592639 .. 11592816 (178

87% 142592 .. 142712 (121) 11592860 .. 11592986 (127

87% 142728 .. 142957 (230) 11593038 .. 11593267 (230

136946 AGGATAACGATGACGTTGCCTGAGCGAACACAATGAAGCTTATAAACGAAATTCAATAGCTGAAAATCGGCGTCCCATCTGCGCTC---------- 137032

>>>

1585773 ATGATGACGATGACGTTGCCTGAACGCACACAATGAAGCTTATAAACGAAATTCAATAGCTGAAAATCGGCGTCCCATCTGTGCTCTTTGGGATTC 1585868

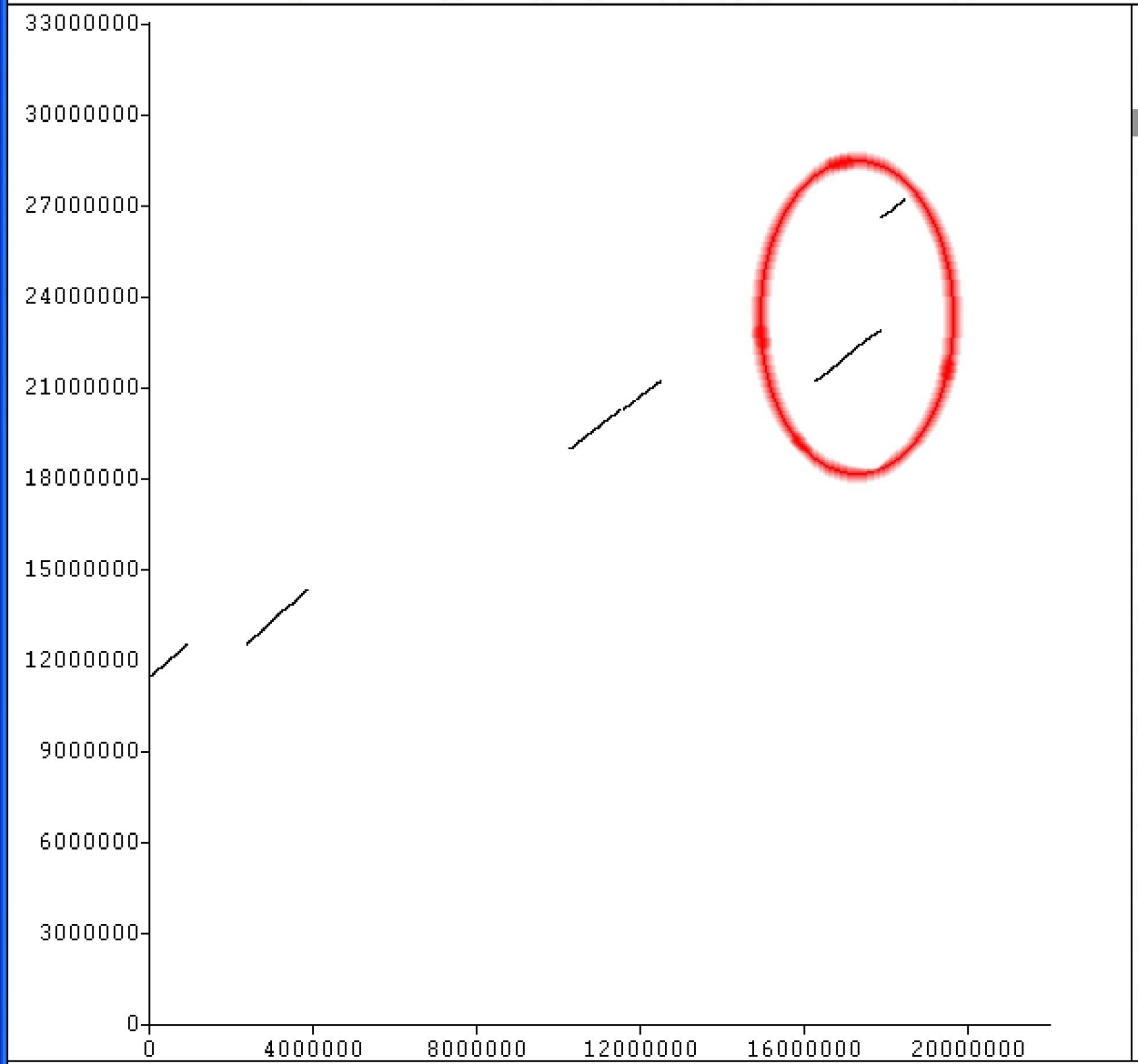
Action: Greedy Deleted: 0 Total: 11707 Conflicting: 0

Owen v1.4

File Construct Conflict Filter Select Edit Window Help

1 .. 20206255 D.Viril scaffold_12963 type=golden_path_region; loc=scaffold_12963:1..20206255; ID=scaffold_12963; dbxref=GB:CH94064

1 .. 32352404 D.Mojaven scaffold_6500 type=golden_path_region; loc=scaffold_6500:1..32352404; ID=scaffold_6500; dbxref=GB:CH9338



88% 136722 .. 136877 (156) 11585441 .. 11585593 (153

86% 136946 .. 137141 (196) 11585773 .. 11585979 (207

78% 137166 .. 137314 (149) 11586093 .. 11586230 (138

85% 137789 .. 137898 (110) 11587018 .. 11587135 (118

84% 137988 .. 138183 (196) 11587256 .. 11587452 (197

86% 138242 .. 138344 (103) 11587538 .. 11587645 (108

83% 138410 .. 138557 (148) 11587753 .. 11587903 (151

84% 138825 .. 139049 (225) 11588378 .. 11588594 (217

89% 139775 .. 140039 (265) 11589342 .. 11589603 (262

81% 140195 .. 140463 (269) 11589698 .. 11589984 (287

85% 140553 .. 140640 (88) 11590135 .. 11590222 (88)

85% 141082 .. 141235 (154) 11590839 .. 11590995 (157

95% 141304 .. 141365 (62) 11591047 .. 11591109 (63)

77% 141390 .. 141484 (95) 11591139 .. 11591239 (101)

79% 141529 .. 141733 (205) 11591256 .. 11591474 (219

82% 142350 .. 142536 (187) 11592639 .. 11592816 (178

87% 142592 .. 142712 (121) 11592860 .. 11592986 (127

87% 142728 .. 142957 (230) 11593038 .. 11593267 (230

136946 AGGATAACGATGACGTTGCCTGAGCGAACACAATGAAGCTTATAAACGAAATTCAATAGCTGAAAATCGGCGTCCCATCTGCGCTC---------- 137032

| ||| |||||||||||||||| || |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| |||| >>>

1585773 ATGATGACGATGACGTTGCCTGAACGCACACAATGAAGCTTATAAACGAAATTCAATAGCTGAAAATCGGCGTCCCATCTGTGCTCTTTGGGATTC 1585868

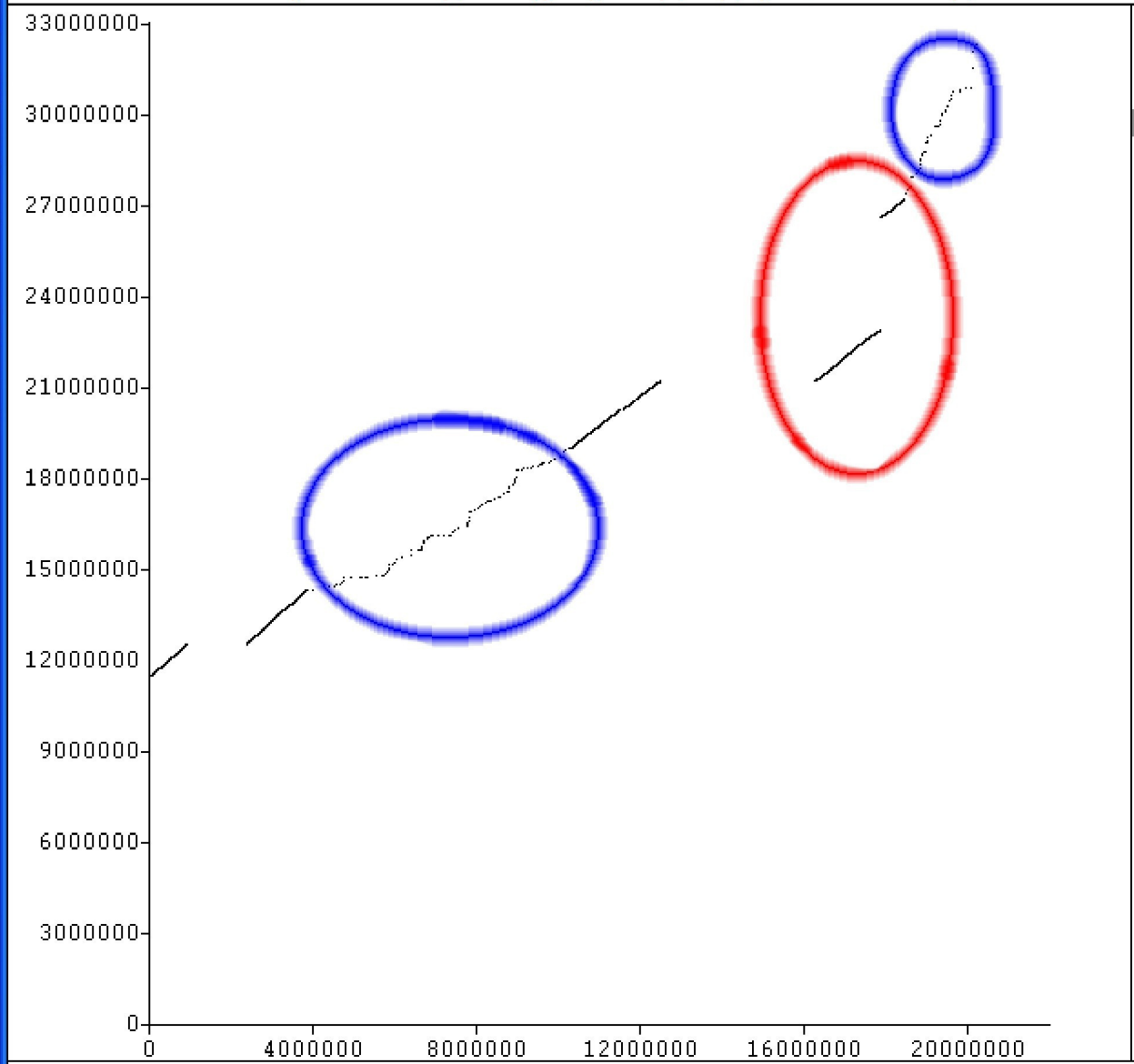
Action: Greedy Deleted: 0 Total: 11707 Conflicting: 0

Owen v1.4

File Construct Conflict Filter Select Edit Window Help

1 .. 20206255 D.Viril scaffold_12963 type=golden_path_region; loc=scaffold_12963:1..20206255; ID=scaffold_12963; dbxref=GB:CH94064

1 .. 32352404 D.Mojaven scaffold_6500 type=golden_path_region; loc=scaffold_6500:1..32352404; ID=scaffold_6500; dbxref=GB:CH9338



88% 136722 .. 136877 (156) 11585441 .. 11585593 (153

86% 136946 .. 137141 (196) 11585773 .. 11585979 (207

78% 137166 .. 137314 (149) 11586093 .. 11586230 (138

85% 137789 .. 137898 (110) 11587018 .. 11587135 (118

84% 137988 .. 138183 (196) 11587256 .. 11587452 (197

86% 138242 .. 138344 (103) 11587538 .. 11587645 (108

83% 138410 .. 138557 (148) 11587753 .. 11587903 (151

84% 138825 .. 139049 (225) 11588378 .. 11588594 (217

89% 139775 .. 140039 (265) 11589342 .. 11589603 (262

81% 140195 .. 140463 (269) 11589698 .. 11589984 (287

85% 140553 .. 140640 (88) 11590135 .. 11590222 (88)

85% 141082 .. 141235 (154) 11590839 .. 11590995 (157

95% 141304 .. 141365 (62) 11591047 .. 11591109 (63)

77% 141390 .. 141484 (95) 11591139 .. 11591239 (101)

79% 141529 .. 141733 (205) 11591256 .. 11591474 (219

82% 142350 .. 142536 (187) 11592639 .. 11592816 (178

87% 142592 .. 142712 (121) 11592860 .. 11592986 (127

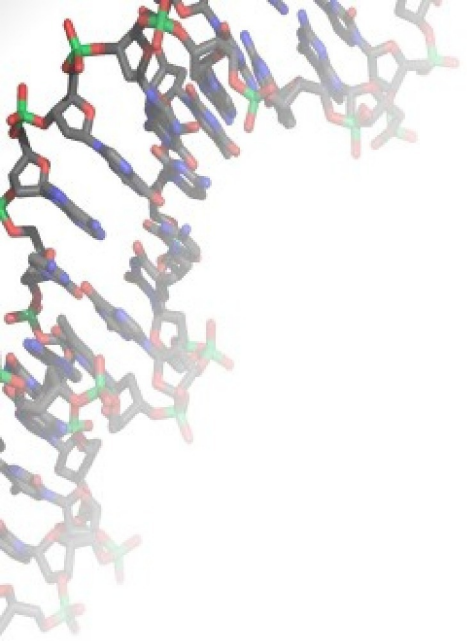
87% 142728 .. 142957 (230) 11593038 .. 11593267 (230

136946 AGGATAACGATGACGTTGCCTGAGCGAACACAATGAAGCTTATAAACGAAATTCAATAGCTGAAAATCGGCGTCCCATCTGCGCTC---------- 137032

| ||| |||||||||||||||| || |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| |||| >>>

1585773 ATGATGACGATGACGTTGCCTGAACGCACACAATGAAGCTTATAAACGAAATTCAATAGCTGAAAATCGGCGTCCCATCTGTGCTCTTTGGGATTC 1585868

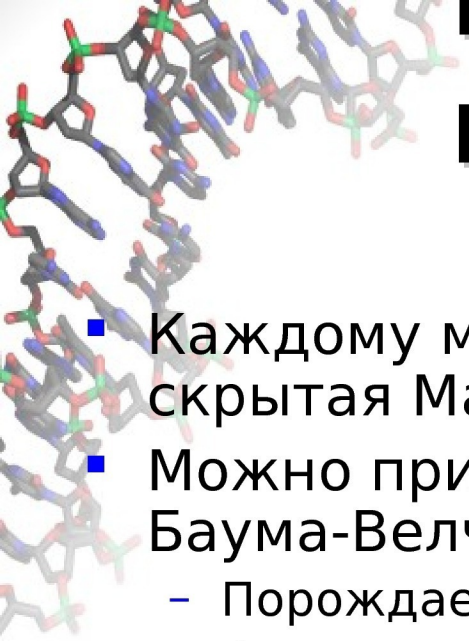
Action: Greedy Deleted: 0 Total: 11707 Conflicting: 0



3. Вес множественного выравнивания – сумма весов столбцов

```
*++++***++
-ИВАН---ОВ----
-ИВАН--ЦОВ----
-ИВАН--КОВ----
-ИВАНЧУКОВ----
ДИВАН---ОВ----
ДИВАНЧИКОВ----
-ИВАН---ОВСКИЙ
--ВАН-ЬКОВ----
```

Недостаток: не учитываем, что «работать» можно не с символами, а с фрагментами.
Вариант: выделяем «значимые» столбцы; все остальное учитываем отдельно в каждой последовательности.



Множественное выравнивание с помощью HMM

- Каждому множественное выравнивание соответствует скрытая Марковская модель.
- Можно применить алгоритм максимизации ожидания Баума-Велча:
  - Порождаем случайные параметры HMM.
  - Выравниваем все последовательности с этой моделью
  - Переоцениваем параметры.
- Проблема: легко попасть в локальный максимум
- Обход проблемы: время от времени параметры HMM возмущаются.
- Другой вариант – использование искусственного отжига.
- Достоинство подхода: одновременно анализируются все последовательности. Нет проблемы необратимости, характерной для прогрессивного выравнивания.