

## アライメント

スコア関数(ギャップを含む)を最大にするような文字の対応つけを探す

1. ギャップなしアライメント
2. ギャップありアライメント

ギャップなし	AFDC AEEC	ギャップあり	AFAED-C A--EEGC
--------	--------------	--------	--------------------

- a. グローバルアライメント (ClustalW)
- b. ローカルアライメント (FASTA, BLAST)

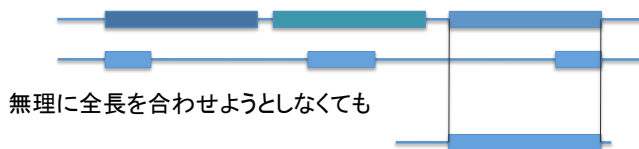
ACDEFGHKLM		ACDEFGHK-LM		FGHK-L
AFGHKKL	➡	A---FGHKKL-	グローバル	FGHKKL
			ローカル	

**動的計画法**というアルゴリズムで解く。  
そのイメージをつかむためには**ドットマトリックス法**が有効

## グローバルアライメントとローカルアライメント

配列の全長が一致すると見なされる場合、グローバルアライメント

たとえばマルチドメインタンパク質(幾つかの要素から構成)と  
シングルドメインのアライメントで



無理に全長を合わせようとしなくても

最も一致度の高い一部分にマッチすれば良い: ローカルアライメント

ACDEFGHK-LM	FGHK-L
A---FGHKKL-	FGHKKL

無理に全長をあわせて  
配列相同性  $6/11 = \sim 55\%$

一致する部分のみで比べて  
配列相同性  $5/6 = \sim 83\%$

## ペアワイズアラインメントのアルゴリズム

ペアワイズアラインメントをおこなうには、さまざまなアラインメントのパターンをすべて作って最もスコアの高いものを選択すればよい、

```

ACDEF      ACDEF-      ACDEF--      .....      ACDEF----
ADEFG      -ADEFG      --ADEFG      .....      ----ADEFG

          ACDEF-      ACDEF--      .....      ACDEF----
          A-DEFG      A--DEFG      .....      A----DEFG

          ACDEF-
          AD-EFG      .....
          .....
          .....
          .....
    
```

単純にすべての組み合わせを考えようとすると、配列長の和の階乗のオーダー  
 たとえば  $100! = 10^{158}$  のオーダーなのでどんなに計算機が速くても無理！  
 実際には、極端に重なりが少ないものを除いたりして計算量を減らすことができるが、それでも配列長がすこし長くなるとすぐに計算は困難になる

計算の仕方の工夫により実現可能な計算量にすることができる！  
 動的計画法 <- この雰囲気をつかむためにまずドットマトリクスをみてみよう

## ドットマトリクス : 例1 (1)

※スコア:一致:+1、不一致:0、ギャップ:-1とする。

配列1

1: GCTAGACTCG

2: AGCTAGACTC

(1) 配列1、配列2を  
横と縦に並べる

		G	C	T	A	G	A	C	T	C	G
配列2 ↓	A										
	G										
	C										
	T										
	A										
	G										
	A										
	C										
	T										
	C										

### ドットマトリックス : 例1 (2)

※スコア:一致:+1、不一致:0、ギャップ:-1とする。

1: GCTAGACTCG  
2: AGCTAGACTC

(1) 配列1、配列2を横と縦に並べる

(2) 文字が一致するマスに○を描く

		配列1 →									
		G	C	T	A	G	A	C	T	C	G
A	↓ 配列2				○		○				
G		○				○					○
C			○					○		○	
T				○					○		
A					○		○				
G		○				○					○
A					○		○				
C			○					○		○	
T				○					○		
C		○						○		○	

### ドットマトリックス : 例1 (3)

※スコア:一致:+1、不一致:0、ギャップ:-1とする。

1: GCTAGACTCG  
2: AGCTAGACTC

(1) 配列1、配列2を横と縦に並べる

(2) 文字が一致するマスに○を描く

(3) 多くの○を通るような左上と右下を結ぶ折れ線

		配列1 →									
		G	C	T	A	G	A	C	T	C	G
A	↓ 配列2				○		○				
G		○				○					○
C			○					○		○	
T				○					○		
A					○		○				
G		○				○					○
A					○		○				
C			○					○		○	
T				○					○		
C		○						○		○	

### ドットマトリックス : 例1 (4)

※スコア:一致:+1、不一致:0、ギャップ:-1とする。

1: GCTAGACTCG  
2: AGCTAGACTC

(1) 配列1、配列2を横と縦に並べる

(2) 文字が一致するマスに○を描く

(3) 多くの○を通るような左上と右下を結ぶ折れ線

(4) アライメント

1: -GCTAGACTCG  
\*\*\*\*\*  
2: AGCTAGACTC-

配列1 →

	G	C	T	A	G	A	C	T	C	G
A				○		○				
G	○				○					○
C		○					○		○	
T			○					○		
A				○		○				
G	○				○					○
A				○		○				
C		○						○		○
T			○					○		
C		○						○		○

配列2 ↓

スコア:一致(+1)×9+不一致(0)×0+ギャップ(-1)×2=7

### ローカルアライメントの解法 (Smith & Waterman, 1981)

(0) 準備  
格子の端のスコアを0に設定

(1) 前向きステップ

$$F(i, j) = \max \begin{cases} F(i+1, j+1) + s(x_i, y_j) & \text{対角}(d) \\ F(i+1, j) + \text{Gap} & \text{鉛直}(v) \\ F(i, j+1) + \text{Gap} & \text{水平}(h) \\ 0 & \text{終結}(0) \end{cases}$$

(2) 後ろ向きステップ  
最大のスコアのノードを探し、そのノードを起点にして辿る。パス'0'が現れたら終了

## ドットマトリックス : 例2 (1)

※スコア:一致:+1、不一致:0、ギャップ:-1とする。

配列1: GCTCGACTTG

配列2: GCACGCTATG

(1) 配列1、配列2を  
横と縦に並べる

		配列1 →									
		G	C	T	C	G	A	C	T	T	G
↓ 配列2	G										
	C										
	A										
	C										
	G										
	C										
	T										
	A										
	T										
	G										

## ドットマトリックス : 例2 (2)

※スコア:一致:+1、不一致:0、ギャップ:-1とする。

配列1: GCTCGACTTG

配列2: GCACGCTATG

(1) 配列1、配列2を  
横と縦に並べる

(2) 文字が一致する  
マスに○を描く

		配列1 →									
		G	C	T	C	G	A	C	T	T	G
↓ 配列2	G	○				○					○
	C		○		○			○			
	A						○				
	C		○		○			○			
	G	○				○					○
	C		○		○			○			
	T			○					○	○	
	A						○				
	T			○					○	○	
	G	○				○					○

## ドットマトリックス : 例2 (3)

※スコア:一致:+1、不一致:0、ギャップ:-1とする。

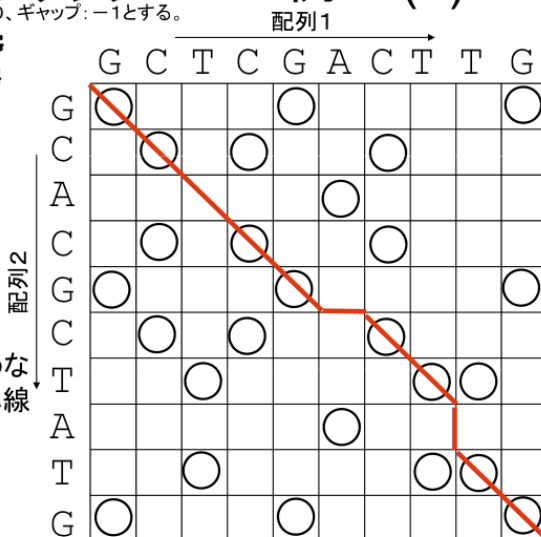
配列1: GCTCGACTTG

配列2: GCACGCTATG

(1) 配列1、配列2を  
横と縦に並べる

(2) 文字が一致する  
マスに○を描く

(3) 多くの○を通るような  
左上と右下を結ぶ折れ線



## ドットマトリックス : 例2 (4)

※スコア:一致:+1、不一致:0、ギャップ:-1とする。

配列1: GCTCGACTTG

配列2: GCACGCTATG

(1) 配列1、配列2を  
横と縦に並べる

(2) 文字が一致する  
マスに○を描く

(3) 多くの○を通るような  
左上と右下を結ぶ折れ線

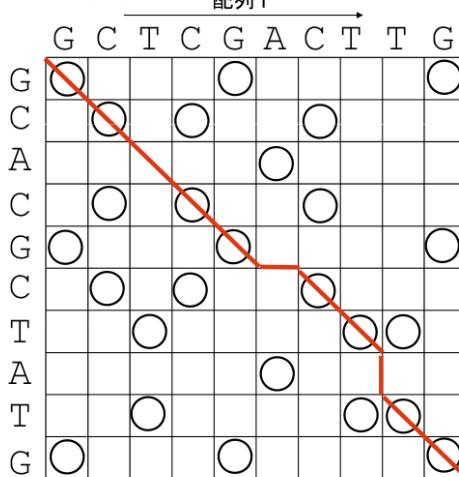
(4) アライメント

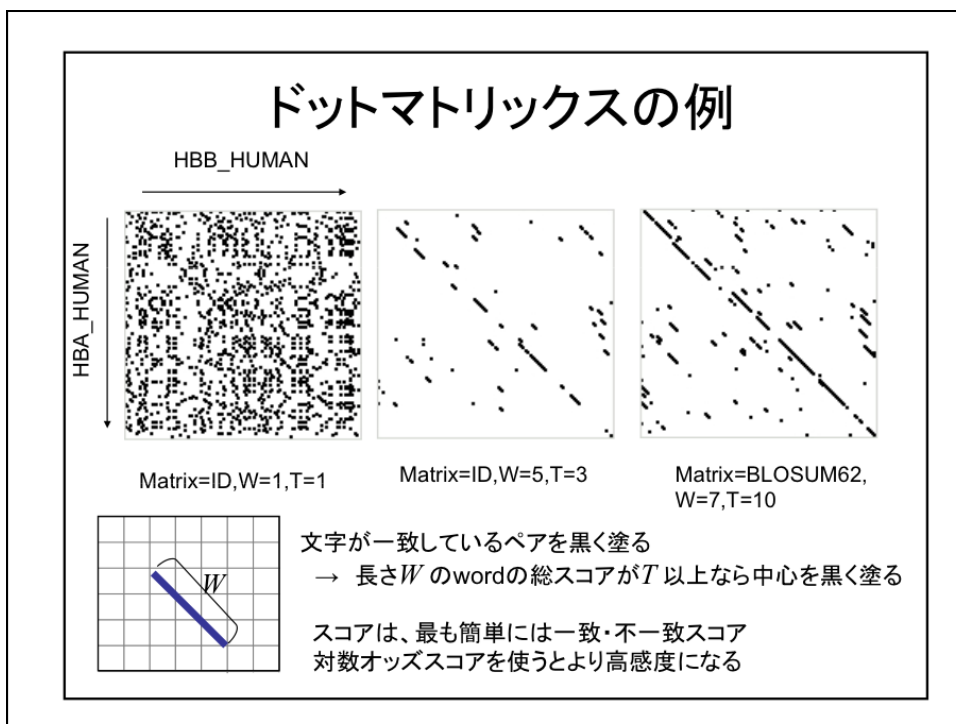
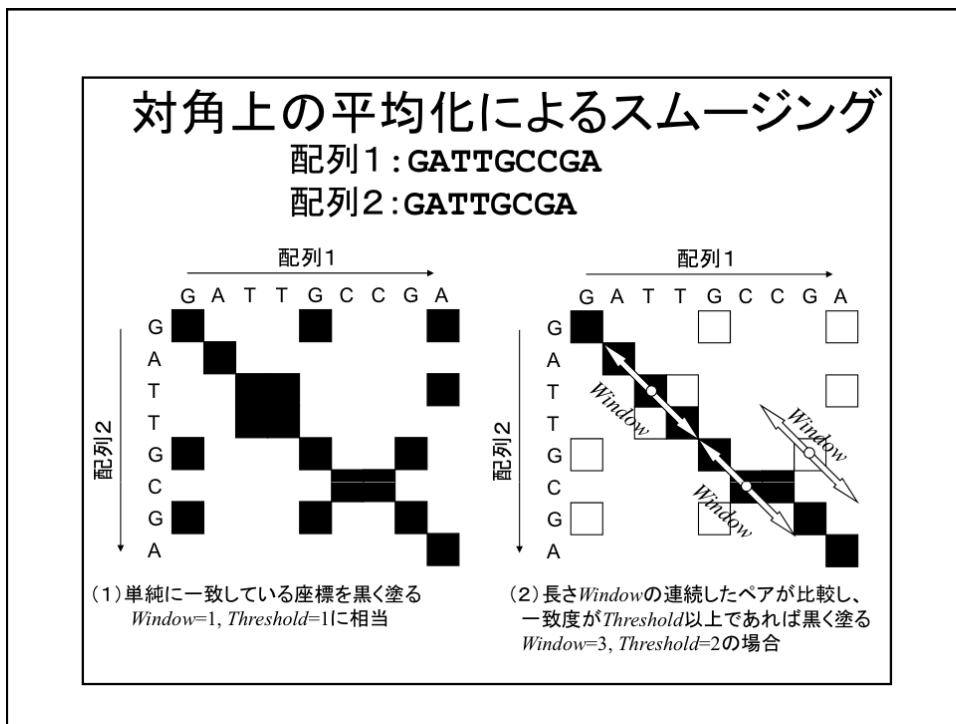
1: GCTCGACT-TG

\*\*\* \*\* \*\* \*\*

2: GCACG-CTATG

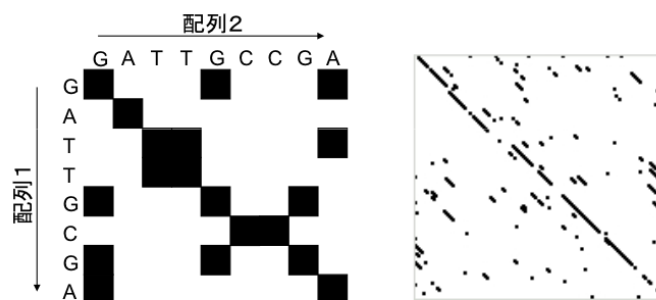
スコア:一致(+1)×8+不一致(0)×1+ギャップ(-1)×2=6





## ドットマトリックス法の特徴

- アルゴリズムが平易
- 非常に長い配列の比較にも対応
- 部分一致、繰り返しなど特殊なケースにも対応できる。
- あくまでグラフィカルな対応なので、具体的な文字列対応(アライメント)は与えない。



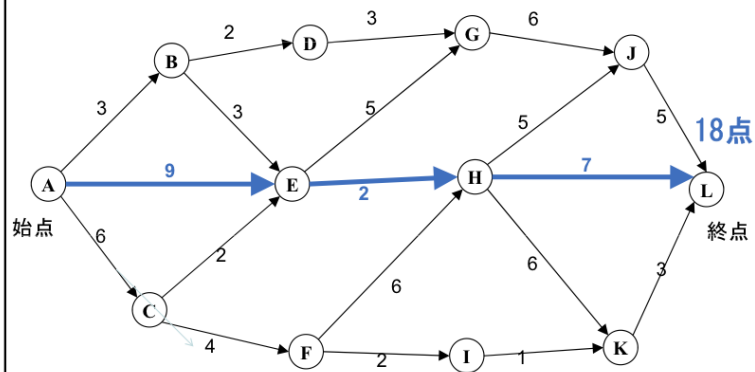
## 動的計画法によるアライメント

- アライメント問題は、**有向グラフの最適経路問題**と等価
- 有向グラフの最適経路問題は**動的計画法** (Dynamic Programming)と呼ばれるアルゴリズムで解ける。
- $O(NM)$ の計算量(文字列長の積に比例)



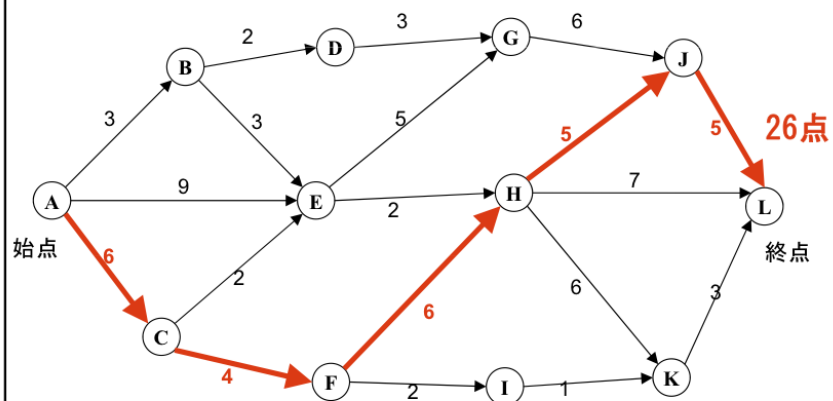
### 最適経路問題

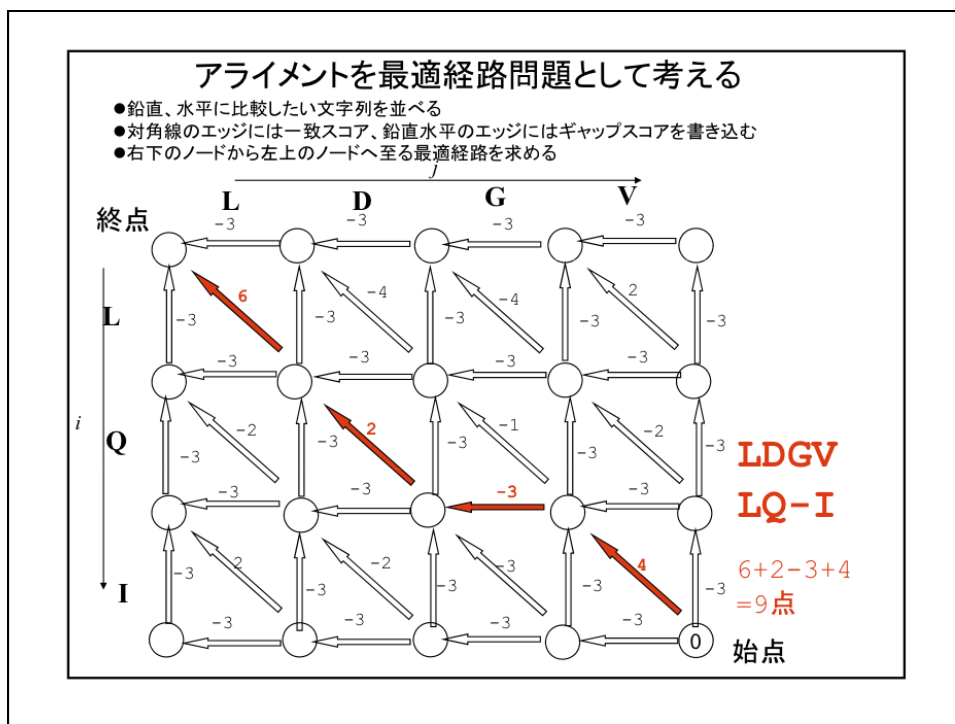
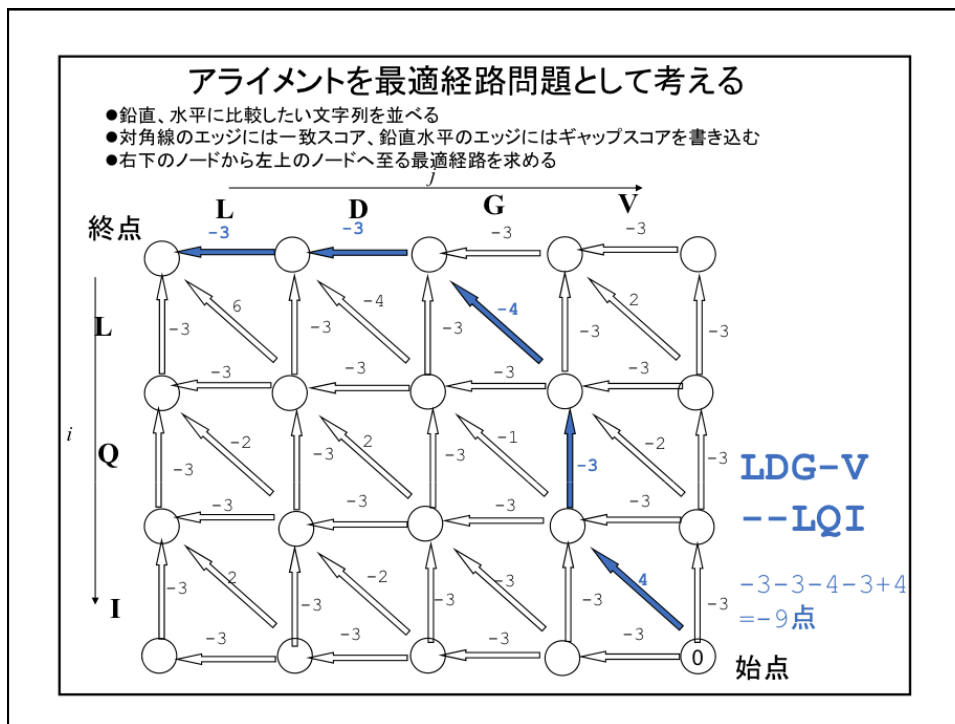
始点Aから終点Lにいたるエッジの得点の合計が最大となる経路を探す



### 最適経路問題

始点Aから終点Lにいたるエッジの得点の合計が最大となる経路を探す

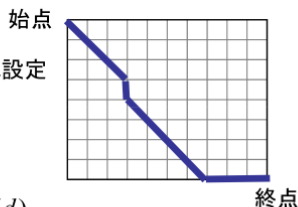




### グローバル・アライメントの解法 (Needleman & Wunsch, 1970)

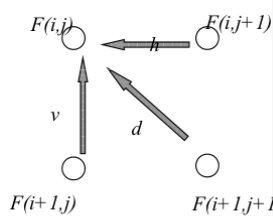
(0)準備

右端の列、下端の行の格子点のスコアを0に設定



(1)前向きステップ

$$F(i, j) = \max \begin{cases} F(i+1, j+1) + S(x_i, y_j) & \text{対角}(d) \\ F(i+1, j) + \text{Gap} & \text{鉛直}(v) \\ F(i, j+1) + \text{Gap} & \text{水平}(h) \end{cases}$$

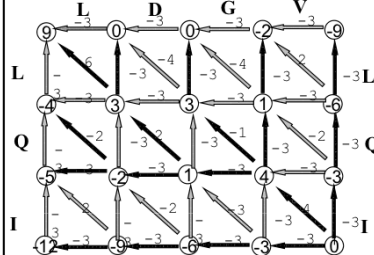


(2)後ろ向きステップ

始点を起点にして辿る。終点に到着したら終了。

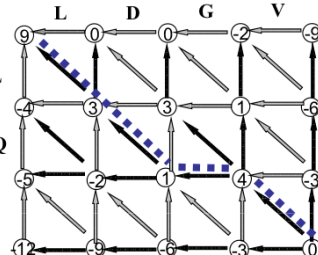
### 動的計画法の手続き

(1)前向き (Forward)

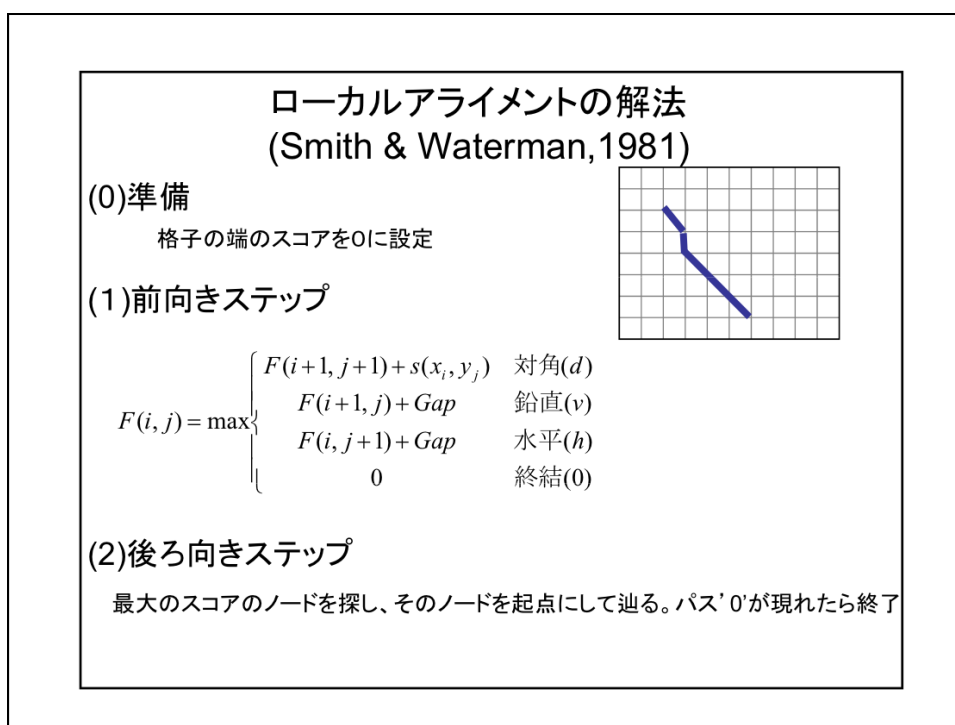
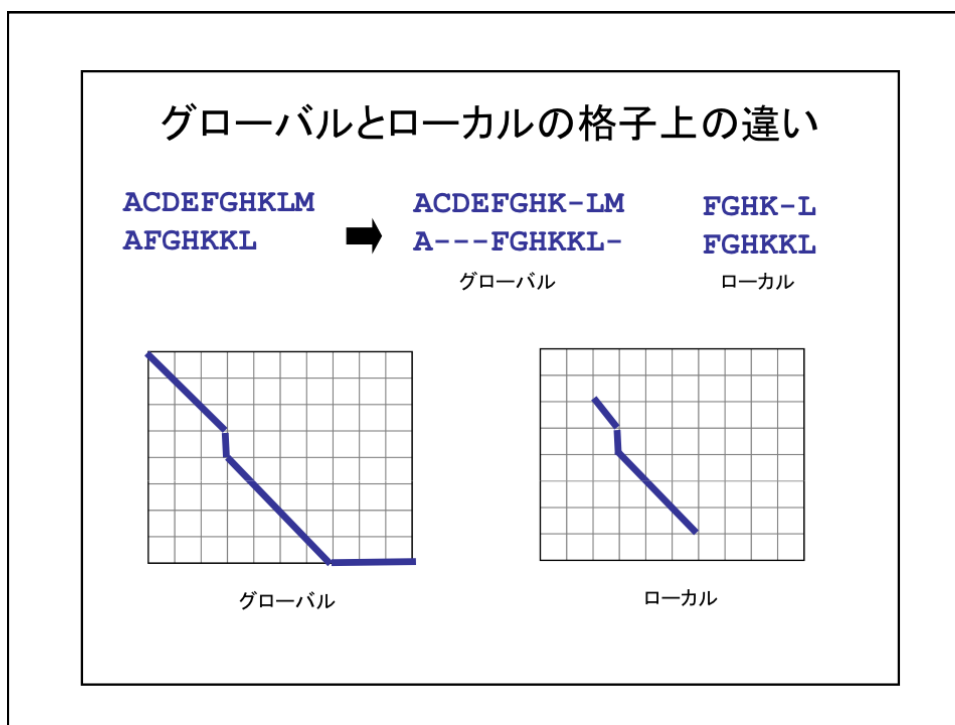


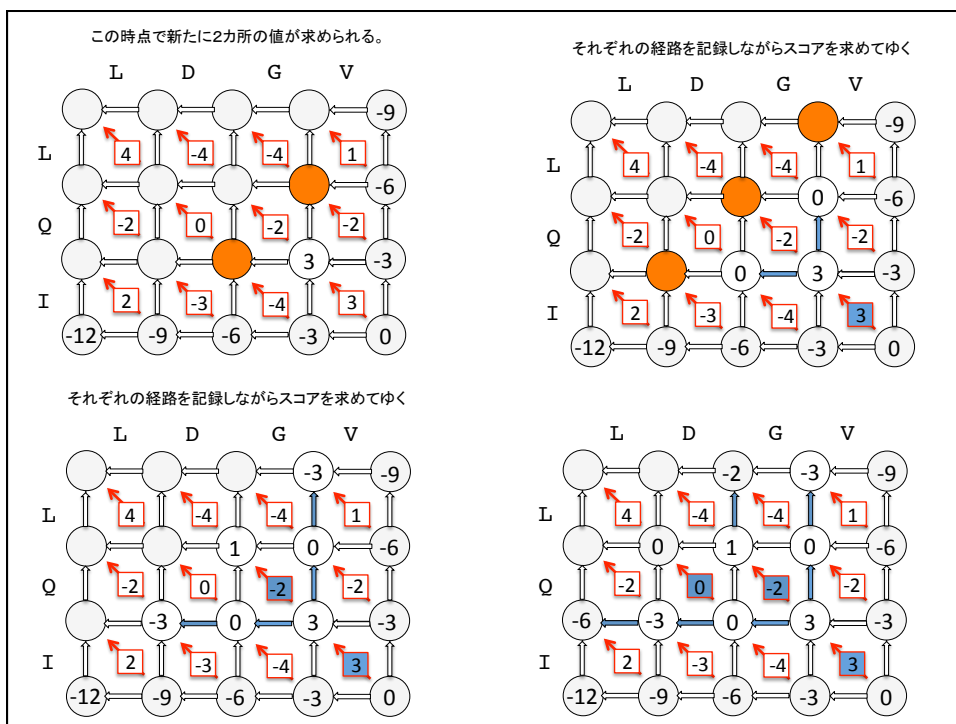
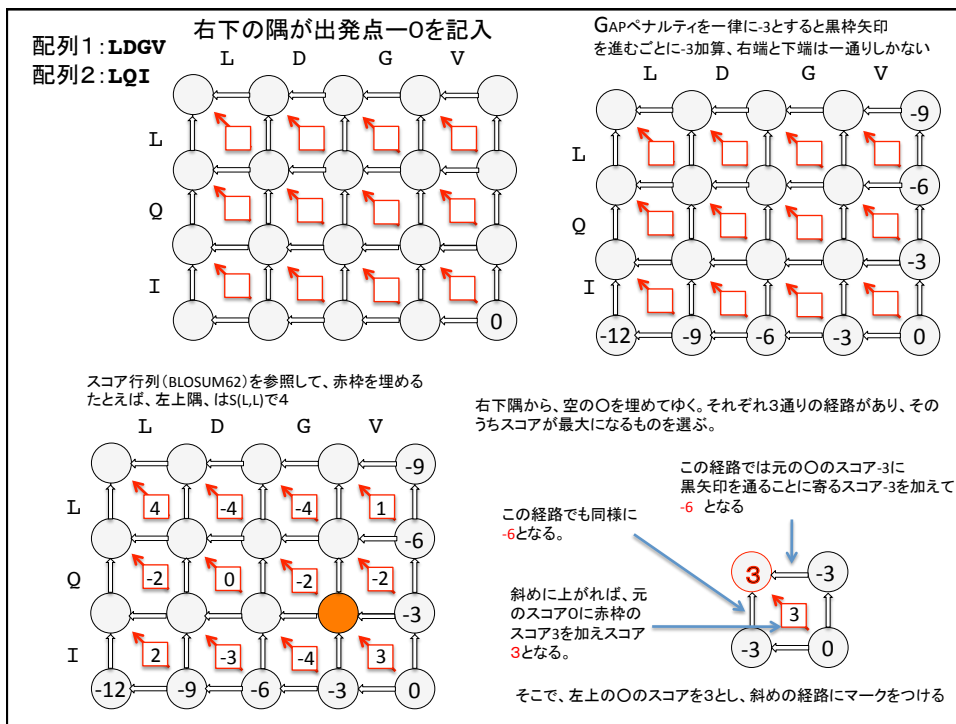
$O(NM)$

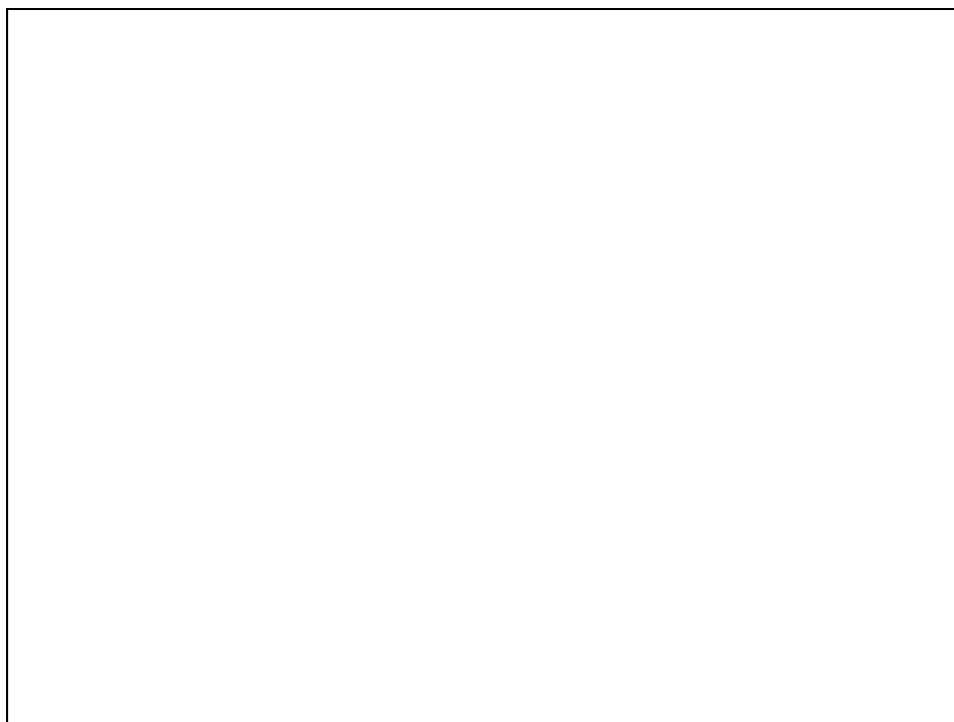
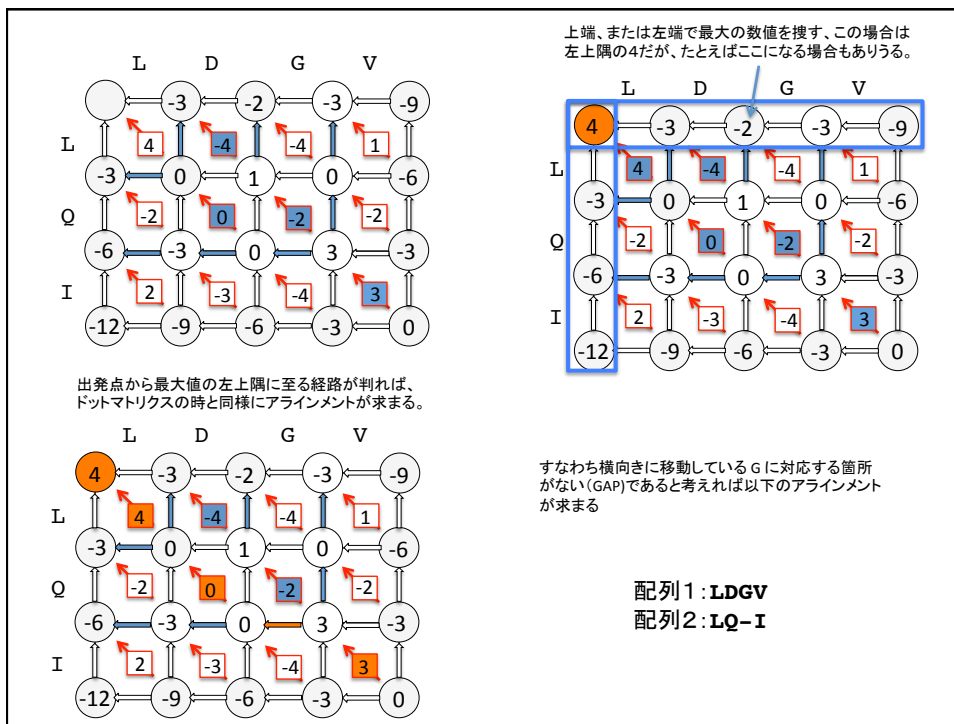
(2)後ろ向き (TraceBack)



LDGV  
LQ-I

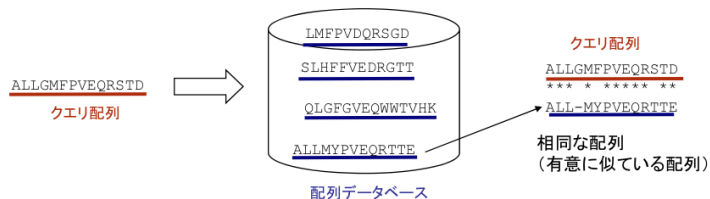






## 配列相同性検索

→クエリ配列を配列データベースと比較、相同な配列を探す



- 機能未知遺伝子の機能予測 (アノテーション)  
機能既知の配列との類似 → 機能の類似を示唆
- 立体構造予測  
構造既知の配列との類似 → 構造の類似を示唆
- 遺伝子発見  
既知遺伝子と類似している領域の発見 → 遺伝子の存在を示唆

## E-value

### 配列相同性のもう一つの指標

後で用いる配列相同性検索プログラム: BLASTで用いられる

ログオッズスコアの和: アラインメントが長いほど高くなる → 補正

ランダムな配列の比較で、偶然にスコアが生じる可能性

0~1で、低いほど、相同性が高いと考えられる

ひとつの目安として、 $0.0001 = 10^{-4}$  より小さければホモロジーが有ると考える

BLASTの出力では、指数表記で表されるので注意

例えば、 $10^{-4}$  は  $1.0e-4$  と表記される

0.24 は  $2.4e-1$ , 0.000000000098 は、 $9.8e-11$  と表記される

## BLASTホームページ

http://blast.ncbi.nlm.nih.gov/Blast.cgi

Google等で、「NCBI BLAST」で検索

The screenshot shows the NCBI BLAST homepage. At the top, there is a navigation bar with 'Home', 'Recent Results', 'Saved Strategies', and 'Help'. Below this, the main heading is 'BLAST Assembled Genomes'. A red box highlights the 'protein blast' option under the 'Basic BLAST' section. A red arrow points from the text 'タンパク質アミノ酸配列で検索' to this option. Another red arrow points from the text 'ここを押して検索実行' to the 'BLAST' button at the bottom of the page.

Annotations on the screenshot:

- タンパク質アミノ酸配列で検索 (Protein sequence search)
- ここを押して検索実行 (Click here to execute search)

## タンパク質アミノ酸配列に対するBLAST検索ページ

The screenshot shows the BLAST search page. The 'Enter Query Sequence' section has a red arrow pointing to the 'From' field with the text 'アミノ酸配列をペースト'. The 'Choose Search Set' section has a red arrow pointing to the 'Database' dropdown menu with the text '検索対象のデータベースを選択' and 'ここではNR'. The 'Program Selection' section has a red arrow pointing to the 'blastp' radio button with the text '検索手法を選択' and 'ここではblastp'. At the bottom, a red arrow points to the 'BLAST' button with the text 'ここを押して検索実行'. Another red arrow points to the 'Algorithm parameters' link with the text 'より細かな検索条件を設定(次ページ)'.

Annotations on the screenshot:

- アミノ酸配列をペースト (Paste amino acid sequence)
- 検索対象のデータベースを選択 (Select search target database)
- ここではNR (Here is NR)
- 検索手法を選択 (Select search method)
- ここではblastp (Here is blastp)
- ここを押して検索実行 (Click here to execute search)
- より細かな検索条件を設定(次ページ) (Set more detailed search conditions (next page))



細かい条件設定のページ(前ページのAlgorithm parametersをクリックした場合)

結果の配列数の上限  
(デフォルトでは100個まで出力)

結果の配列数の上限  
(デフォルトでは100個まで出力)

スコアマトリクスを選択

ギャップペナルティを選択

検索の実行

タンパク質アミノ酸配列に対するBLAST検索ページ

UNIPROTから  
HBA\_HUMAN  
の配列をとってきて  
貼り付けた

ここを押して検索実行

### BLAST検索結果

クエリー配列長の領域における予測機能

クエリー配列(~140)の全長にわたって高い相同性の配列が高い横棒の数だけあるここでは上限の100本

下にスクロール(次ページ)

#### 一行あたり一つのタンパク質

それぞれのE-value すべて10のマイナス76~67乗オーダー = すべてホモログ

Accession	Description	Score (Bits)	E Value
gb AAK29522.1	hemoglobin alpha 2 [synthetic construct]	287	3e-76
ref NP_000508.1	alpha 2 globin [Homo sapiens] >ref NP_000549.1	286	4e-76
pdb 3IA3 B	Chain B, A Cis-Proline In Alpha-Hemoglobin Stabili...	286	4e-76
gb AAK04486.1	hemoglobin alpha-2 [Homo sapiens]	285	8e-76
gb AAK72612.1 AF230076.1	alpha-2-globin [Homo sapiens]	285	1e-75
gb ABF56145.1	hemoglobin alpha 1-2 hybrid [Homo sapiens]	284	2e-75
pdb 1BAB A	Chain A, Hemoglobin Thionville: An Alpha-Chain Var...	284	2e-75
pdb 1ABY A	Chain A, Cyanomet Rhb1.1 (Recombinant Hemoglobin) ...	284	2e-75
gb AAK37554.1 AF349571.1	hemoglobin alpha-1 globin chain (Hom...	284	2e-75
pdb 1C7D A	Chain A, Deoxy Rhb1.2 (Recombinant Hemoglobin)	284	2e-75
pdb 1COH A	Chain A, Structure Of Haemoglobin In The Deoxy Qua...	284	2e-75
gb 1BAB97112.1	alpha 2 globin variant [Homo sapiens]	284	3e-75
gb AAK91973.1	alpha-2 globin [Pongo pygmaeus] >gb AAK29174.1	283	3e-75
pdb 1BZ2 A	Chain A, Hemoglobin (Alpha V1m) Mutant >pdb 1BZ2[C...	283	5e-75
pdb 1A91 A	Chain A, R-State Human Carbonmonoxyhemoglobin Alph...	283	5e-75
sp F01923.1 HBA_GORGO	RecName: Full=Hemoglobin subunit alpha;...	282	8e-75
pdb 1O1O A	Chain A, Deoxy Hemoglobin (A,C,vim,V621; B,D,vim,V...	281	1e-74
pdb 1ZH5 A	Chain A, Solution Structure Of Human Normal Adult ...	281	1e-74
pdb 1Y0D A	Chain A, T-To-Thigh Quaternary Transitions In Huma...	281	1e-74
pdb 1A3O A	Chain A, Artificial Mutant (Alpha Y42h) Of Deoxy H...	281	1e-74
pdb 1O1N A	Chain A, Deoxy Hemoglobin (A-Glyglygly-C,vim,L29w;...	281	1e-74
sp Q9P935.2 HBA1_HV1A	RecName: Full=Hemoglobin subunit alpha;...	281	1e-74
pdb 1YDZ A	Chain A, T-To-T(High) Quaternary Transitions In Hu...	281	2e-74
pdb 1RW1A	Chain A, R State Human Hemoglobin (alpha V96w). Ca...	281	2e-74
gb ACE60606.1	hemoglobin alpha chain [Hipposideros armiger]	258	2e-67
sp P28780.1 HBA_TAFGE	RecName: Full=Hemoglobin subunit alpha;...	257	3e-67
gb ACE60603.1	hemoglobin alpha chain [Chaerephon plicatus]	257	3e-67
sp P20854.1 HBA_CTRGN	RecName: Full=Hemoglobin subunit alpha;...	257	3e-67
gb ACC62116.1	theta 1 globin (predicted) [Rhinolophus ferrum...	256	3e-67
gb ACE60605.1	hemoglobin alpha chain [Eonycteris spelaea]	256	4e-67
sp I29839.1 HBA_MKCCA	RecName: Full=Hemoglobin subunit alpha;...	256	4e-67
sp F01953.1 HBA_MELMF	RecName: Full=Hemoglobin subunit alpha;...	256	5e-67
sp F01936.1 HBA_EULFU	RecName: Full=Hemoglobin subunit alpha;...	256	5e-67
sp Q862A7.3 HBA_PIFAR	RecName: Full=Hemoglobin subunit alpha;...	256	6e-67

下にスクロール(次ページ)

### クエリーとヒットのアラインメント

**1本目**

```
>|cbl|AAK29522.1| hemoglobin alpha 2 [synthetic construct]
Length=143
Score = 287 bits (734), Expect = 3e-76, Method: Compositional matrix adjust.
Identities = 142/142 (100%), Positives = 142/142 (100%), Gaps = 0/142 (0%)
Query 1  MVLSPADKTNVKAAMKGVGAHAGEYGAELERFLSPFTTKTYFPHFDLSHGSAQVKGHG 60
Sbjct 1  MVLSPADKTNVKAAMKGVGAHAGEYGAELERFLSPFTTKTYFPHFDLSHGSAQVKGHG 60
Query 61  KKVADALTNAAVHDDMPNALSALSDLHAHLKRVDPVNFKLLSHCLLVTLAAHLPAEFTF 120
KKVADALTNAAVHDDMPNALSALSDLHAHLKRVDPVNFKLLSHCLLVTLAAHLPAEFTF 120
Sbjct 61  KKVADALTNAAVHDDMPNALSALSDLHAHLKRVDPVNFKLLSHCLLVTLAAHLPAEFTF 120
Query 121 AVHASLDKFLASVSTVLTSEYR 142
AVHASLDKFLASVSTVLTSEYR
Sbjct 121 AVHASLDKFLASVSTVLTSEYR 142
```

一つ目のヒットに関する情報

完全一致

**最後**

```
>|cbl|086227.3|HBA_F1PAB RecName: Full=Hemoglobin subunit alpha; AltName: Full=Hemoglobin
alpha chain; AltName: Full=Alpha-globin
db|BAC57967.1| alpha globin [Pipistrellus abramus]
Length=143
Score = 256 bits (653), Expect = 6e-67, Method: Compositional matrix adjust.
Identities = 128/143 (89%), Positives = 132/143 (92%), Gaps = 1/143 (0%)
Query 1  MVLSPADKTNVKAAMKGVGAHAGEYGAELERFLSPFTTKTYFPHFDLSHGSAQVKGHG 59
Sbjct 1  MVLSPADKTNVKAAMKGVGAHAGEYGAELERFLSPFTTKTYFPHFDLSHGSAQVKGHG 60
Query 60  GKVDALTNAAVHDDMPNALSALSDLHAHLKRVDPVNFKLLSHCLLVTLAAHLPAEFT 119
GKVDALTNAAVHDDMPNALSALSDLHAHLKRVDPVNFKLLSHCLLVTLAAHLPAEFT 119
Sbjct 61  GKVDALTNAAVHDDMPNALSALSDLHAHLKRVDPVNFKLLSHCLLVTLAAHLPAEFT 120
Query 120 FAVHASLDKFLASVSTVLTSEYR 142
FAVHASLDKFLASVSTVLTSEYR
Sbjct 121 FAVHASLDKFLANVSTVLTSEYR 143
```

配列相同性 89%

アブラコウモリ