

Introduction to Dynamic Programming

The sequence alignment problem

Wilson Leung 12/27/2025

1

Outline

- ⊗ Overview of the sequence alignment problem
- ⊗ Calculate the optimal global alignment
- ⊗ Characteristics of dynamic programming algorithms
- ⊗ Calculate the optimal local alignment

2

Learning objectives

- ⊗ Understand the theory behind sequence alignment
 - ⊗ Become a **better informed user** of NCBI BLAST
- ⊗ This presentation will not cover:
 - ⊗ The BLAST algorithm
 - ⊗ Parameter optimizations
 - ⊗ Statistics for similarity searches (Karlin-Altschul theory)

Korf, I., Yandell, M., and Bedell, J. (2003). BLAST. O'Reilly Media, Inc.

3

Design goals

Drosophila melanogaster legless (lgs), mRNA
Sequence ID: ref|NM_143665.4| Length: 5357 Number of Matches: 6

Score	Expect	Identities	Gaps	Strand	Next Match	Previous Match
2762 bits(3062)	0.0	1822/2016(90%)	6/2016(0%)	Plus/Minus		
Query 3359	ATTACCAGCAGAGGACTGACCGAATGAGTCCAATTCCTTTGGTGATAGATGGGATAGAGG	3418				
Sbjct 3209	ATCACCAGCAGAGGACTGACCGAAAAGACTCAAATTCCTTTGGGGATAGATGAGATAAGGG	3150				
Query 3419	GGTACTTGGGTTGCTGCTTAAGTTATGCGTAAAGACGCTTGATGATCGGGTATTCTACT	3478				
Sbjct 3149	GGTACTTGGGTTGCTGCTTAAGTTATGCGTAAAGACGCTTGACGATCCGGTATTCTACT	3090				

- ⊗ Generate an alignment between **two sequences**
- ⊗ Identify the “best” (**most parsimonious**) alignment
- ⊗ Generate the best alignment “**quickly**”

4

Strategy #1: Visual inspection

```

Query:  ATTACCAG
        ||| |||
Subject: ATCACCAG
  
```

- ⊗ Sequences must have **high percent identity**
- ⊗ Applications:
 - ⊗ PAM scoring matrix (align sequences with $\geq 85\%$ identity)
 - ⊗ Align mononucleotide runs during sequence improvement

5

Strategy #2: Enumerate all alignments

- ⊗ Guaranteed to find the best alignment
- ⊗ Does not scale
 - ⊗ Combinatorial explosion
 - ⊗ Two 300 bp sequences have $\sim 10^{179}$ possible alignments (Eddy 2004)
- ⊗ **Brute-force** algorithm
 - ⊗ Establish baseline performance and test cases
 - ⊗ Identify patterns in the problem space

6

Apply the brute force algorithm to a single column of the alignment

Homologous Not homologous

Query:

A

A-	-A
----	----

Subject:

A

-A	A-
----	----

- Three possible alignments for two 1 bp sequences
- Query length (M) = 1; Subject length (N) = 1
- Only two **biological interpretations**:
 - A in the query is **homologous** to A in the subject
 - A in the query is **not homologous** to A in the subject

7

Six possible relationships between the query and subject for M=2, N=2

2 aligned bases	1 aligned base		0 aligned bases							
Query: <table border="1"><tr><td>A</td><td>T</td></tr></table>	A	T	<table border="1"><tr><td>A-T</td></tr></table>	A-T	<table border="1"><tr><td>-AT</td></tr></table>	-AT	<table border="1"><tr><td>AT--</td></tr></table>	AT--	<table border="1"><tr><td>--AT</td></tr></table>	--AT
A	T									
A-T										
-AT										
AT--										
--AT										
Subject: <table border="1"><tr><td>A</td><td>T</td></tr></table>	A	T	<table border="1"><tr><td>-AT</td></tr></table>	-AT	<table border="1"><tr><td>A-T</td></tr></table>	A-T	<table border="1"><tr><td>--AT</td></tr></table>	--AT	<table border="1"><tr><td>AT--</td></tr></table>	AT--
A	T									
-AT										
A-T										
--AT										
AT--										
	<table border="1"><tr><td>AT-</td></tr></table>	AT-	<table border="1"><tr><td>A-T</td></tr></table>	A-T	<table border="1"><tr><td>A-T-</td></tr></table>	A-T-	<table border="1"><tr><td>-A-T</td></tr></table>	-A-T		
AT-										
A-T										
A-T-										
-A-T										
	<table border="1"><tr><td>A-T</td></tr></table>	A-T	<table border="1"><tr><td>AT-</td></tr></table>	AT-	<table border="1"><tr><td>-A-T</td></tr></table>	-A-T	<table border="1"><tr><td>A-T-</td></tr></table>	A-T-		
A-T										
AT-										
-A-T										
A-T-										
	<table border="1"><tr><td>AT-</td></tr></table>	AT-	<table border="1"><tr><td>-AT</td></tr></table>	-AT	<table border="1"><tr><td>A--T</td></tr></table>	A--T	<table border="1"><tr><td>-AT-</td></tr></table>	-AT-		
AT-										
-AT										
A--T										
-AT-										
	<table border="1"><tr><td>-AT</td></tr></table>	-AT	<table border="1"><tr><td>AT-</td></tr></table>	AT-	<table border="1"><tr><td>-AT-</td></tr></table>	-AT-	<table border="1"><tr><td>A--T</td></tr></table>	A--T		
-AT										
AT-										
-AT-										
A--T										

Each color denotes a different **evolutionary relationship**

8

Observations from the brute force alignment strategy

- Many of the possible alignments are **redundant**
 - Imply the same evolutionary relationship
- Large number** of possible alignments
 - 13 possible alignments for sequences of length 2
- Can **ignore** many possible alignments
 - Many are suboptimal compared to the best alignment

9

Strategy #3: Dot plot

- Cell position (i, j):
 - i = Query position (x-axis)
 - j = Subject position (y-axis)
- Draw a dot** at (i, j) if the two bases are **identical**
- Connect the dots** to make a line (alignment)
- Level of noise depends on repeat density
 - Use **longer words** and higher cutoff scores to reduce noise

10

Assessment of the three sequence alignment strategies

- Infeasible to examine all possible alignments
 - Need to reduce the search space
- Only a small subset of alignments are “interesting”
 - Many alignments are redundant
- Connect the dots in the dot plot to create an alignment
 - Consider the **cumulative levels of similarity**

11

The optimal alignment is composed of smaller optimal alignments

Query:

A	T
---	---

 Subject:

A	T
---	---

Query:

A	T
---	---

A	-	T
---	---	---

A	T	-
---	---	---

Subject:

A	T
---	---

-	A	T
---	---	---

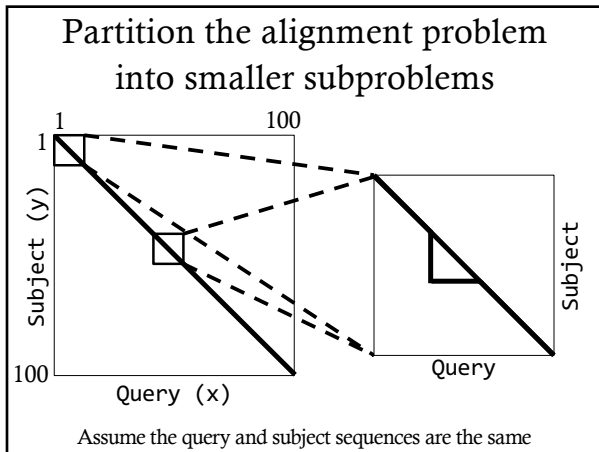
A	-	T
---	---	---

- Only the best alignment** at each position could be part of the final optimal alignment

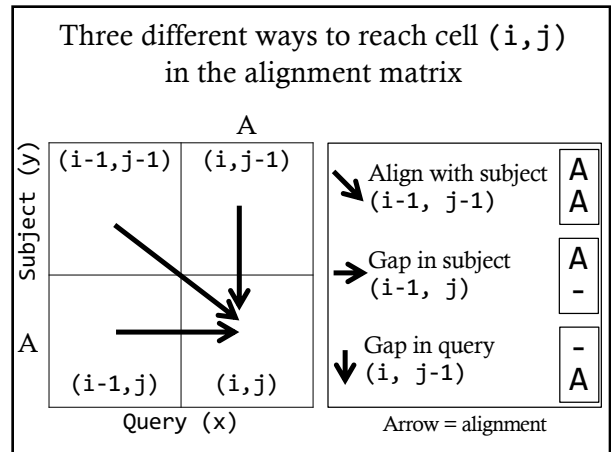
A	-	T	-
-	A	-	T

■ Align ■ Deletion in subject ■ Insertion in subject

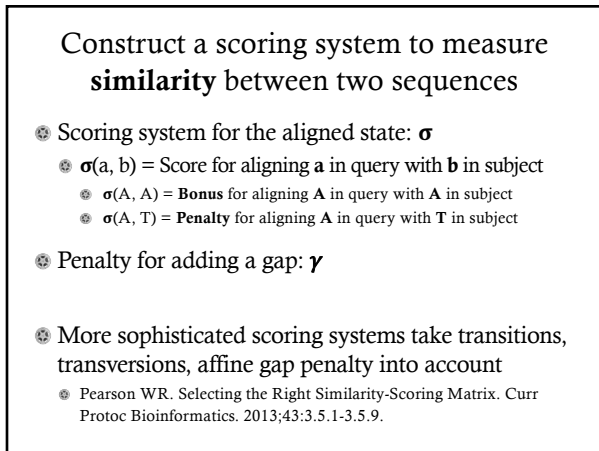
12



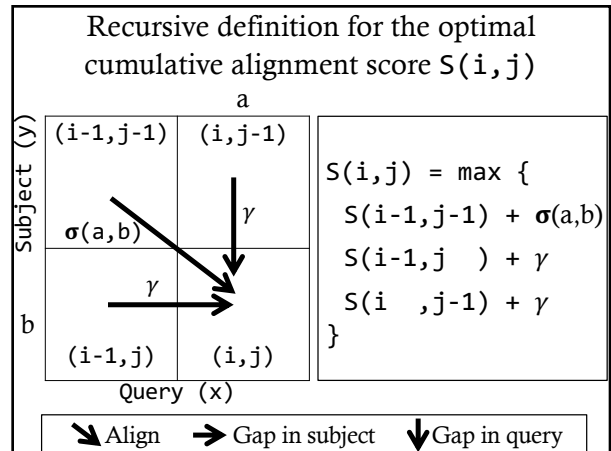
13



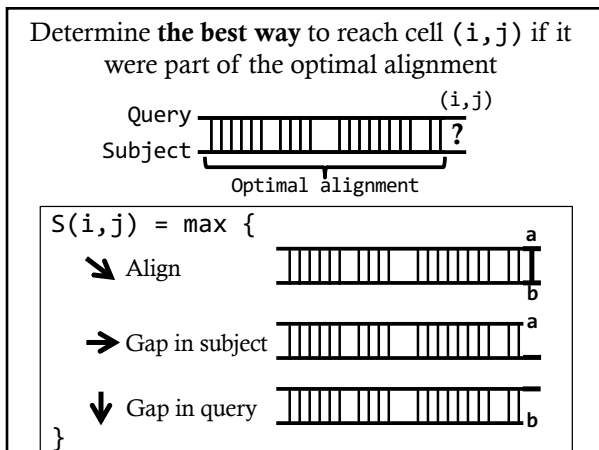
14



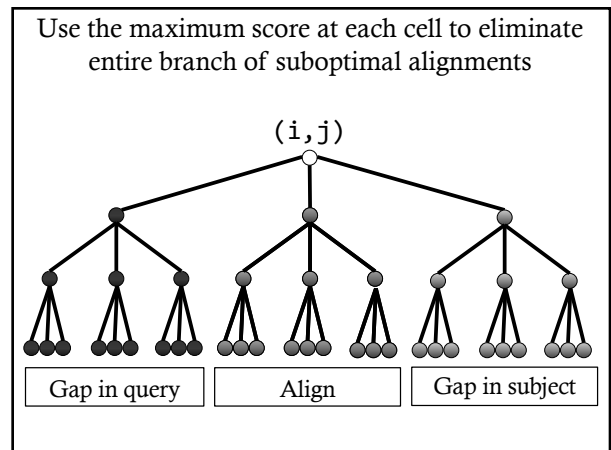
15



16



17



18

Cumulative score $S(i, j)$ encapsulates the alignment decisions up to position (i, j)

- All potential optimal alignments that go through cell (i, j) have the same ancestry
 - Re-use the cumulative alignment score (**memoization**)
- Gaps are described by the cumulative score
 - Do not affect the coordinates of the alignment matrix
- Do not know the optimal alignment until we complete the entire alignment matrix
 - Optimal alignment has the **highest cumulative score**

19

Needleman-Wunsch algorithm (global alignment)
(Query length: M ; Subject length: N)

- Construct a $(M+1) \times (N+1)$ matrix
 - Extra column and row = gaps at the beginning of the alignment
- Fill in the cells in the first row and first column with the cumulative gap costs
- Calculate the **maximum score** for subsequent cells (i, j)
 - Keep track of the decision that leads to the maximum score (S)

$$S(i, j) = \max \begin{cases} S(i-1, j-1) + \sigma(a, b) \\ S(i-1, j) + \gamma \\ S(i, j-1) + \gamma \end{cases}$$

Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol. 1970 Mar;48(3):443-53.

20

Initialize the alignment matrix
(Match = +5; Mismatch = -2; Gap = -6)

	0	1	2	3	4	5	6	7	8
		T	G	C	T	C	G	T	A
0	0	-6	-12	-18	-24	-30	-36	-42	-48
1	T	-6							
2	T	-12							
3	C	-18							
4	A	-24							
5	T	-30							
6	A	-36							

Query (Eddy, 2004)

21

Calculate the possible scores for the cell at position $(1, 1)$

		T	
	$(0, 0)$	$(1, 0)$	
Subject (y)	0	-6	
	$\sigma(T, T)$	γ	
T	-6	$(1, 1)$	
	$(0, 1)$		
		γ	

Query (x)

$$S(1, 1) = \max \begin{cases} S(0, 0) + \sigma(T, T) \\ S(0, 1) + \gamma \\ S(1, 0) + \gamma \end{cases}$$

22

Calculate the optimal score for the cell at position $(1, 1)$

		T	
	0	-6	
Subject (y)	+5	-6	
T	-6	5	-12
		-6	
		-12	5

Query (x)

$$S(1, 1) = \max \begin{cases} 0 + (+5) = 5 \\ -6 + (-6) = -12 \\ -6 + (-6) = -12 \end{cases}$$

$S(1, 1) = 5$

(Match = +5; Mismatch = -2; Gap = -6)

23

Calculate the possible scores for the cell at position $(2, 1)$

		T	G
	$(1, 0)$	$(2, 0)$	
Subject (y)	-6	-12	
	$\sigma(T, G)$	γ	
T	5	$(2, 1)$	
	$(1, 1)$		
		γ	

Query (x)

$$S(2, 1) = \max \begin{cases} S(1, 0) + \sigma(T, G) \\ S(1, 1) + \gamma \\ S(2, 0) + \gamma \end{cases}$$

24

Calculate the optimal score for the cell at position (2,1)

		T	G	
Subject (y)	T	-6	-12	
	T	5	-8	-18
	Query (x)			

(Match = +5; Mismatch = -2; Gap = -6)

$$S(2,1) = \max \{$$

- 6 + (-2) = -8
- 5 + (-6) = -1
- 12 + (-6) = -18

$$S(2,1) = -1$$

25

Alignment matrix after two iterations
(Match = +5; Mismatch = -2; Gap = -6)

		0	1	2	3	4	5	6	7	8
		0	T	G	C	T	C	G	T	A
Subject	0	0	-6	-12	-18	-24	-30	-36	-42	-48
	1	T	-6	5	-1					
	2	T	-12							
	3	C	-18							
	4	A	-24							
	5	T	-30							
	6	A	-36							

Query

26

Calculate the optimal score for the cell at position (3,1)

		G	C	
Subject (y)	T	-12	-18	
	T	-1	-14	-24
	Query (x)			

(Match = +5; Mismatch = -2; Gap = -6)

$$S(3,1) = \max \{$$

- 12 + (-2) = -14
- 1 + (-6) = -7
- 18 + (-6) = -24

$$S(3,1) = -7$$

27

Matrix after three iterations
(Match = +5; Mismatch = -2; Gap = -6)

		0	1	2	3	4	5	6	7	8
		0	T	G	C	T	C	G	T	A
Subject	0	0	-6	-12	-18	-24	-30	-36	-42	-48
	1	T	-6	5	-1	-7				
	2	T	-12							
	3	C	-18							
	4	A	-24							
	5	T	-30							
	6	A	-36							

Query

28

Calculate the optimal score for the cell at position (1,2)

			T	
Subject (y)	T	-6	5	
	T	-12	-1	-1
	Query (x)			

(Match = +5; Mismatch = -2; Gap = -6)

$$S(1,2) = \max \{$$

- 6 + (+5) = -1
- 12 + (-6) = -18
- 5 + (-6) = -1

$$S(1,2) = -1$$

29

Complete alignment matrix
(Match = +5; Mismatch = -2; Gap = -6)

		0	1	2	3	4	5	6	7	8	
		0	T	G	C	T	C	G	T	A	
Subject	0	0	-6	-12	-18	-24	-30	-36	-42	-48	
	1	T	-6	5	-1	-7	-13	-19	-25	-31	-37
	2	T	-12	-1	3	-3	-2	-8	-14	-20	-26
	3	C	-18	-7	-3	8	2	3	-3	-9	-15
	4	A	-24	-13	-9	2	6	0	1	-5	-4
	5	T	-30	-19	-15	-4	7	4	-2	6	0
	6	A	-36	-25	-21	-10	1	5	2	0	11

Query

30

Use **traceback** to recover the optimal alignment

- ⊛ Start from the cell within the last row and last column that has the highest score
- ⊛ **Recall the step (color)** that leads to this optimal score
 - ⊛ Report this step in the alignment output
 - ⊛ **All the alignment decisions have already been made**
- ⊛ Repeat until we reached the beginning of the sequence
- ⊛ Two options if multiple paths produce the same score
 - ⊛ Report only one of the paths (pick arbitrarily)
 - ⊛ Report all paths with the optimal score

31

Query: T C G T A
 Subject: T C A T A
 Traceback:

Query	T	G	C	T	C	G	T	A
0	-6	-12	-18	-24	-30	-36	-42	-48
T	-6	5	-1	-7	-13	-19	-25	-31
T	-12	-1	3	-3	-2	-8	-14	-20
C	-18	-7	-3	8	2	-3	-9	-15
A	-24	-13	-9	2	6	0	1	-5
T	-30	-19	-15	-4	7	4	-2	6
A	-36	-25	-21	-10	1	5	2	0
A								11

32

Calculate the optimal score for the cell at position (5, 3)

Subject (y)	T	C
T	-2	-8
C	2	3

Query (x): T C

(Match = +5; Mismatch = -2; Gap = -6)

$$S(5,3) = \max \{$$

- 2 + (+5) = 3
- 2 + (-6) = -4
- 8 + (-6) = -14

$$S(5,3) = 3$$

33

Traceback must follow the steps that produce the optimal **cumulative** global alignment score

Subject (y)	T	C
T	-2	-8
C	2	3

Query (x): T C

34

Query: T G C T C G T A
 Subject: T - - T C A T A
 Traceback:

Query	T	G	C	T	C	G	T	A
0	-6	-12	-18	-24	-30	-36	-42	-48
T	-6	5	-1	-7	-13	-19	-25	-31
T	-12	-1	3	-3	-2	-8	-14	-20
C	-18	-7	-3	8	2	-3	-9	-15
A	-24	-13	-9	2	6	0	1	-5
T	-30	-19	-15	-4	7	4	-2	6
A	-36	-25	-21	-10	1	5	2	0
A								11

35

The Needleman-Wunsch algorithm is an example of a **dynamic programming** algorithm

- ⊛ Problem must satisfy two criteria:
 - ⊛ **Optimal substructure**
 - ⊛ Optimal solution to the complete problem is composed of optimal solutions to the subproblems
 - ⊛ **Overlapping problems**
 - ⊛ **Re-use the results** for the subproblems (e.g., lookup table)
- ⊛ Many bioinformatics problems satisfy these criteria
 - ⊛ Sequence alignment, gene prediction, RNA-folding

Bellman B. The theory of dynamic programming. Bulletin of the American Mathematical Society. 1954; 60(6):503-516

36

Smith-Waterman algorithm (local alignment)

(Query length: M; Subject length: N)

- Three changes to the Needleman-Wunsch algorithm:
 - The minimum score for a cell is **zero**
 - Initiate a new alignment when the cumulative score is negative
 - Begin traceback from the cell within **the entire matrix** that has the highest score
 - Terminate traceback when the score is zero

$$S(i,j) = \max \begin{cases} S(i-1,j-1) + \sigma(a,b) \\ S(i-1,j) + \gamma \\ S(i,j-1) + \gamma \\ 0 \end{cases}$$

Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol. 1981 Mar 25;147(1):195-7.

37

Global versus local alignments

- Global alignment**
 - Optimal alignment along the entire length of two sequences
 - Compare protein sequences to identify orthologs
- Local alignment**
 - Optimal alignment between parts of two sequences
 - Identify conserved domains within protein sequences
- Glocal (semi-global) alignment**
 - Optimal global alignment for one sequence; optimal local alignment for the other sequence
 - Map a coding exon against a genomic sequence

38

Initialize the local alignment matrix

(Match = +5; Mismatch = -2; Gap = -6)

	0	1	2	3	4	5	6	7	8	
		T	G	C	T	C	G	T	A	
0	0	0	0	0	0	0	0	0	0	Subject
1	T	0								
2	T	0								
3	C	0								
4	A	0								
5	T	0								
6	A	0								
										Query

39

Calculate the possible local alignment scores for the cell at position (1,1)

			T	
	(0,0)		(1,0)	$S(1,1) = \max \{$ $S(0,0) + \sigma(T,T)$ $S(0,1) + \gamma$ $S(1,0) + \gamma$ 0 $\}$
Subject (y)	0		0	
	$\sigma(T,T)$		γ	
	T	0		(1,1)
				Query (x)

↘ Align → Gap in subject ↓ Gap in query

40

Calculate the optimal local alignment score for the cell at position (1,1)

			T	
	0		0	$S(1,1) = \max \{$ $0 + (+5) = 5$ $0 + (-6) = -6$ $0 + (-6) = -6$ 0 $\}$ $S(1,1) = 5$
Subject (y)	0		0	
	+5		-6	
	T	0		5
				-6
				5
				Query (x)

(Match = +5; Mismatch = -2; Gap = -6)

41

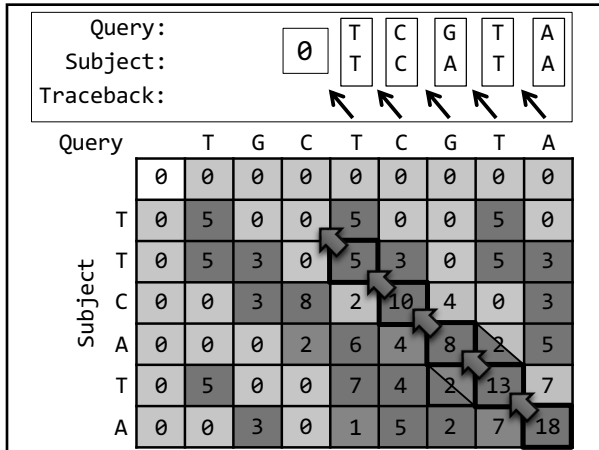
Local alignment matrix

(Match = +5; Mismatch = -2; Gap = -6)

	0	1	2	3	4	5	6	7	8	
		T	G	C	T	C	G	T	A	
0	0	0	0	0	0	0	0	0	0	Subject
1	T	0	5	0	0	5	0	0	5	
2	T	0	5	3	0	5	3	0	5	
3	C	0	0	3	8	2	10	4	0	
4	A	0	0	0	2	6	4	8	2	
5	T	0	5	0	0	7	4	2	13	
6	A	0	0	3	0	1	5	2	7	
										Query

↘ Align → Gap in subject ↓ Gap in query

42

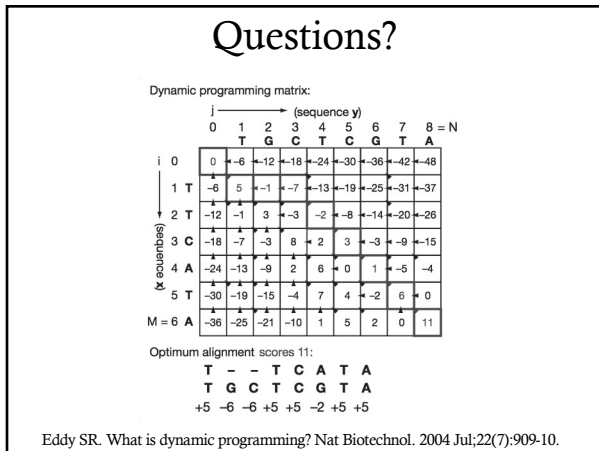


43

Techniques to improve the performance of sequence alignment

- ⊛ Time and space complexity: $O(MN)$
 - ⊛ **Double the size** of the two sequences leads to a **four-fold increase** in the amount of time and space required
- ⊛ Reduce memory requirement
 - ⊛ Myers EW, Miller W. Optimal alignments in linear space. *Comput Appl Biosci.* 1988 Mar;4(1):11-7.
- ⊛ Fill the matrix in parallel (SIMD, CUDA)
 - ⊛ Farrar M. Striped Smith-Waterman speeds database searches six times over other SIMD implementations. *Bioinformatics.* 2007 Jan 15;23(2):156-61.
- ⊛ Find **high-scoring** instead of the best alignment
 - ⊛ Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol.* 1990 Oct 5;215(3):403-10.

44



45



46

Rationale for calculating the scores for the entire alignment matrix

- ⊛ Cannot determine the best global alignment without aligning the entire query and subject sequences
 - ⊛ Cannot evaluate all possible alignments
- ⊛ If the alignment before we reached cell (i, j) is part of the optimal alignment:
 - ⊛ Identify **the next step** (i.e., align, gap in query, gap in subject) that will be part of the optimal alignment
- ⊛ Use **traceback** to determine the final alignment
 - ⊛ Different alignments could produce the same score

47

Overview of the BLAST algorithm

- ⊛ Heuristic algorithm to find **local** regions of similarity between the query and subject sequences
- ⊛ Consists of four main stages:
 - ⊛ Find common subsequences (**words**)
 - ⊛ Extend the word matches into longer alignments
 - ⊛ Evaluate the significance of the high-scoring segment pairs (**HSPs**)
 - ⊛ Combine multiple HSPs into a longer alignment

Korf, I., Yandell, M. and Bedell, J. (2003). The BLAST Algorithm. In *BLAST* (76-87). Sebastopol, CA: O'Reilly Media, Inc.

48

Number of alignments for two sequences with length N

$$f(M,N) = \binom{M+N}{N}$$

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

$$\binom{2N}{N} = \frac{(2N)!}{N!(2N-N)!} = \frac{(2N)!}{(N!)^2}$$

Stirling's approximation
 $n! \approx \sqrt{2\pi n} \left(\frac{n}{e}\right)^n$

$$\lim_{N \rightarrow \infty} \binom{2N}{N} \approx \frac{\left(\sqrt{2\pi(2N)} \left(\frac{2N}{e}\right)^{(2N)}\right)}{\left(\sqrt{2\pi N} \left(\frac{N}{e}\right)^N\right)^2}$$

49

Number of alignments for two sequences with length N

$$\lim_{N \rightarrow \infty} \binom{2N}{N} = \frac{\left(\sqrt{2\pi(2N)} \left(\frac{2N}{e}\right)^{(2N)}\right)}{\left(\sqrt{2\pi N} \left(\frac{N}{e}\right)^N\right)^2} = \frac{\left(\sqrt{2} \sqrt{2\pi N} (2N)^{(2N)} e^{(-2N)}\right)}{\left(\sqrt{2\pi N} (N^N) (e^{-N})\right)^2}$$

$$= \frac{\left(\sqrt{2} \sqrt{2\pi N} (2N)^{(2N)} e^{(-2N)}\right)}{\left(\sqrt{2\pi N}\right)^2 N^{(2N)} e^{(-2N)}}$$

$$= \frac{\left(\sqrt{2} \sqrt{2\pi N}\right)}{\left(\sqrt{2\pi N}\right)\left(\sqrt{2\pi N}\right)} (2^{(2N)}) \left(\frac{(N^{(2N)})(e^{(-2N)})}{(N^{(2N)})(e^{(-2N)})}\right)$$

50

Number of alignments for two sequences with length N

$$= \frac{\cancel{\left(\sqrt{2} \sqrt{2\pi N}\right)}}{\left(\sqrt{2\pi N}\right)\cancel{\left(\sqrt{2\pi N}\right)}} (2^{(2N)}) \left(\frac{(N^{(2N)})(e^{(-2N)})}{(N^{(2N)})(e^{(-2N)})}\right)$$

$$= \left(\frac{1}{\sqrt{\pi N}}\right) (2^{(2N)})$$

$$\lim_{N \rightarrow \infty} \binom{2N}{N} \approx \frac{2^{2N}}{\sqrt{\pi N}}$$

51

Brute force alignment approach is computationally intractable

Sequence length (N)	# possible alignments
10	1.87E+05
50	1.01E+29
100	9.07E+58
200	1.03E+119
300	1.35E+179
400	1.88E+239
500	2.70E+299

$$\lim_{N \rightarrow \infty} \binom{2N}{N} \approx \frac{2^{2N}}{\sqrt{\pi N}}$$

52