

Smith-Waterman Algorithm and Hardware implementation

R08943017 陳傳諭

Outline

- Pairwise Alignment
 - Overview
 - Simple scoring scheme
 - Example
- Local Alignment
 - Smith-Waterman Algorithm
 - Example
 - Affine Gap
- Hardware Implementation

Pairwise Alignment

Sequence *A* : CTTAACT

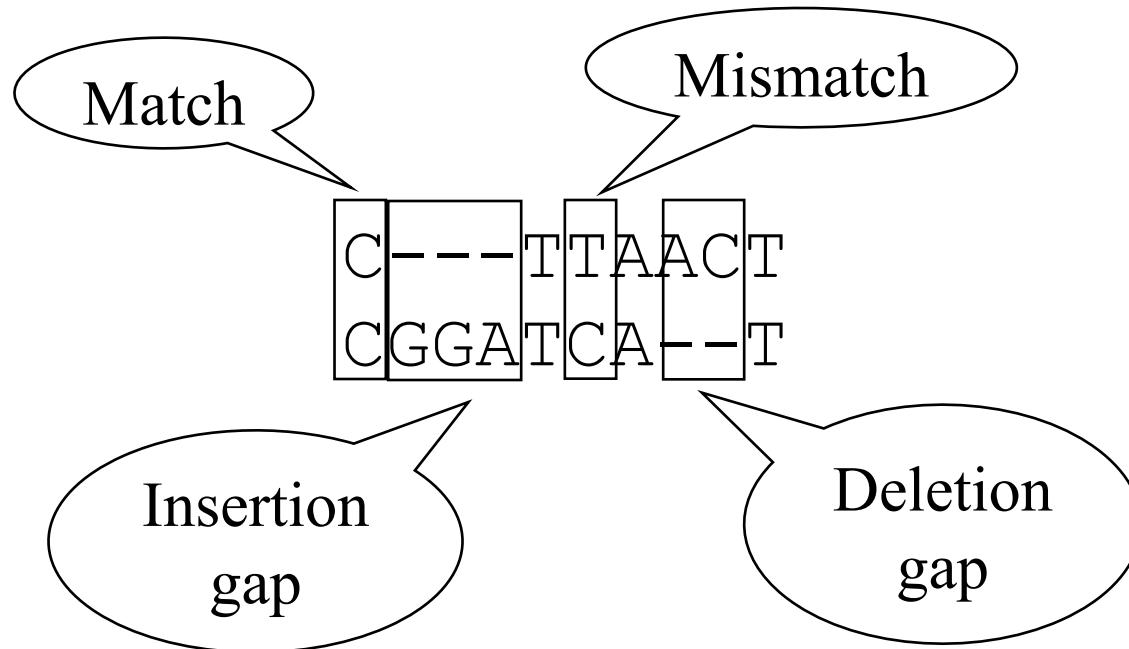
Sequence *B* : CGGATCAT

An alignment of *A* and *B* :

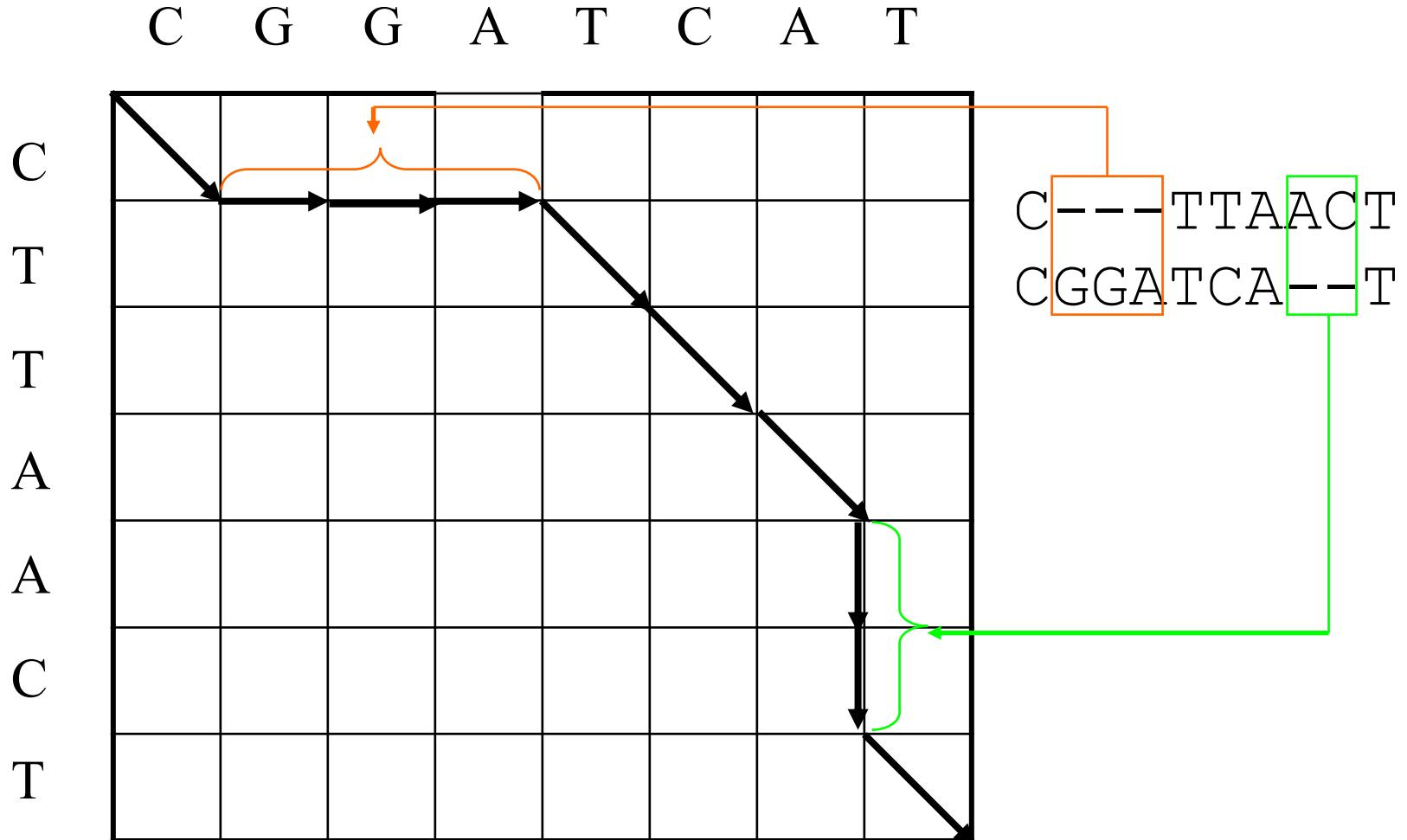
C---TTAACT ← Sequence *A*
CGGATCA--T ← Sequence *B*

Pairwise Alignment

An alignment of A and B :



Alignment Graph



Scoring Scheme

- Match = +8
- Mismatch = -5
- Each Gap = -3

C	-	-	-	T	T	A	A	C	T	
C	G	G	A	T	C	A	-	-	T	
+8	-3	-3	-3	+8	-5	+8	-3	-3	+8	= +12

Scoring Scheme

- Let $A = a_1a_2\dots a_m$ and $B = b_1b_2\dots b_n$
- $S_{i,j}$: the score of an optimal alignment between $a_1a_2\dots a_i$ and $b_1b_2\dots b_j$

$$S_{i,j} = \max \begin{cases} S_{i-1,j} + w(a_i, -) & \text{Gap Penalty} \\ S_{i,j-1} + w(-, b_j) & \\ S_{i-1,j-1} + w(a_i, b_j) & \text{Match / Mismatch} \end{cases}$$

Example

Match : 8

C A A T - T G A

Mismatch : -5

G A A T C T G C

Gap penalty : -3

G A A T C T G C

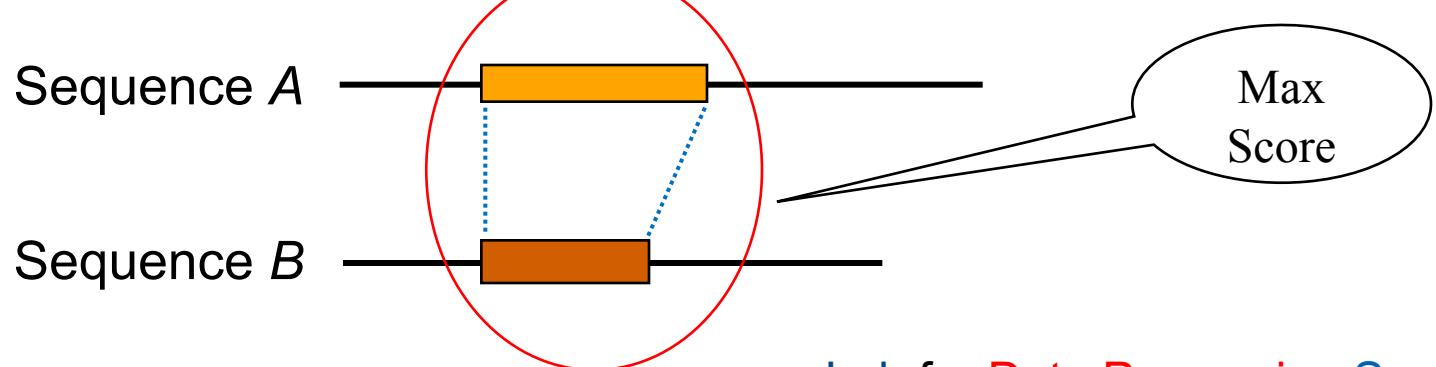
	0	-3	-6	-9	-12	-15	-18	-21	-24
C	-3	-5	-8	-11	-14	-4	-7	-10	-13
A	-6	-8	3	0	-3	-6	-9	-12	-15
A	-9	-11	0	11	8	5	2	-1	-4
T	-12	-14	-3	8	19	16	13	10	7
T	-15	-17	-6	5	16	14	24	21	18
G	-18	-7	-9	2	13	11	21	32	29
A	-21	-10	1	-1	10	8	18	29	27

Local Alignment

- Global Alignment



- Local Alignment



Smith-Waterman Algorithm

- Let $A = a_1a_2\dots a_m$ and $B = b_1b_2\dots b_n$
- $S_{i,j}$: the score of an optimal alignment between $a_1a_2\dots a_i$ and $b_1b_2\dots b_j$

$$S_{i,j} = \max \begin{cases} 0 \\ S_{i-1,j} + w(a_i, -) \\ S_{i,j-1} + w(-, b_j) \\ S_{i-1,j-1} + w(a_i, b_j) \end{cases}$$

Initial Condition

Match : 8

Mismatch : -5

Gap penalty : -3

	G	A	A	T	C	T	G	C
C	0							
A	0							
A	0							
T	0							
T	0							
G	0							
A	0							

Example

Match : 8

Mismatch : -5

Gap penalty : -3

	G	A	A	T	C	T	G	C
C	0	0	0	0	0	0	0	0
A	0	0	0	0	0	8	5	2
A	0	0	8	8	5	5	3	0
A	0	0	8	16	13	10	7	4
T	0	0	5	13	24	21	18	15
T	0	0	2	10	21	$24-5=19$ $21-3=18$ $21-3=18$		
G								
A								

Example

Match : 8

Mismatch : -5

Gap penalty : -3

		G	A	A	T	C	T	G	C
		0	0	0	0	0	0	0	0
C		0	0	0	0	0	8	5	2
A		0	0	8	8	5	5	3	0
A		0	0	8	16	13	10	7	4
T		0	0	5	13	24	21	18	15
T		0	0	2	10	21	19	29	26
G		0	8	5	7	18	16	26	37
A		0	5	16	13	15	13	23	34
									32

Affine Gap

- Match : +8
- Mismatch : -5
- ***Gap Open*** : -7
- Gap extension : -3

C	-	-	-	T	T	A	A	C	T
C	G	G	A	T	C	A	-	-	T
+8	-7	-3	-3	+8	-5	+8	-7	-3	+8 = +4



Affine Gap Penalty

$$D(i, j) = \max \begin{cases} D(i-1, j) - \text{gap_ext} \\ S(i-1, j) - \text{gap_open} \end{cases}$$

$$I(i, j) = \max \begin{cases} I(i, j-1) - \text{gap_ext} \\ S(i, j-1) - \text{gap_open} \end{cases}$$

$$S(i, j) = \max \begin{cases} S(i-1, j-1) + w(a_i, b_j) \\ D(i, j) \\ I(i, j) \\ 0 \end{cases}$$

Hardware Implementation

Hardware Implementation

- Sequence : S & T
- Gap open penalty = α
- Gap extension penalty = β

Hardware Implementation

- Rewrite the equation :

$$E(i, j) = \max \begin{cases} E(i, j - 1) - \beta \\ V(i, j - 1) - \alpha \end{cases}$$

- Initial Condition

$$F(i, j) = \max \begin{cases} F(i - 1, j) - \beta \\ V(i - 1, j) - \alpha \end{cases}$$

- $E^{*,0} = -\infty$

$$V(i, j) = \max \begin{cases} V(i - 1, j - 1) + \sigma(S_i, T_j) \\ E(i, j) \\ F(i, j) \\ 0 \end{cases}$$

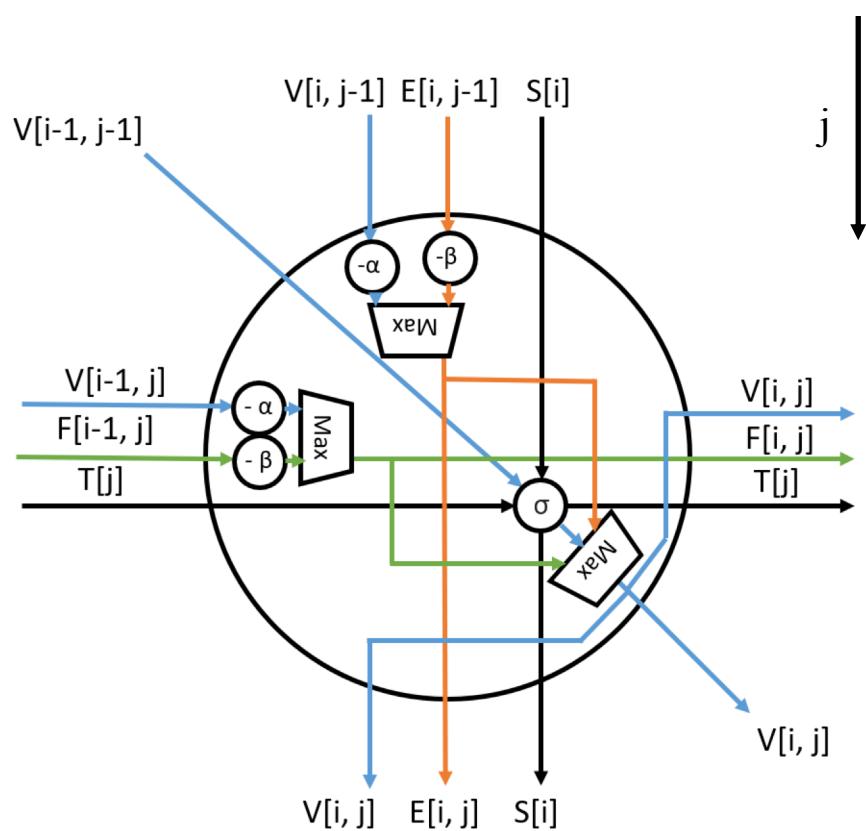
- $F[0,*] = -\infty$

- $V[0,*] = 0$

- $V^{*,0} = 0$

➤ $E^{*,0}, F[0,*]$ can be assigned as 0 as well

PE design



$i \rightarrow$

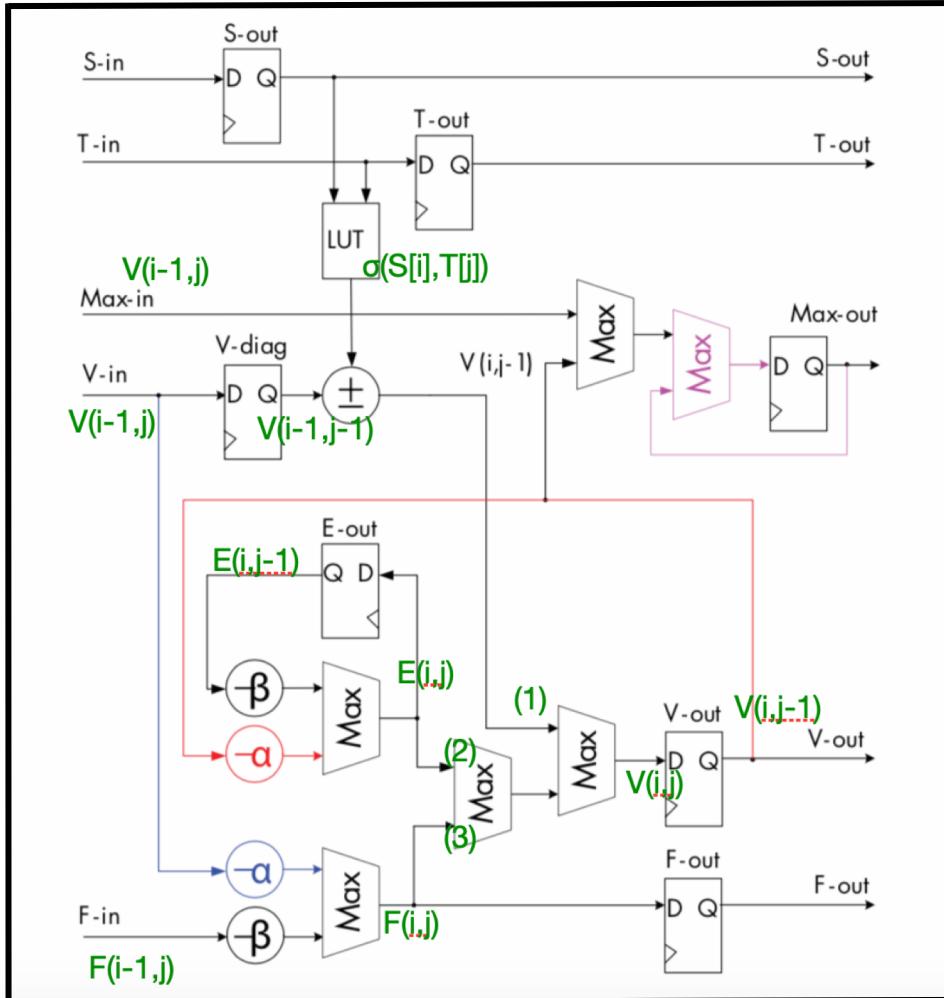
$j \downarrow$

$$E(i, j) = \max \begin{cases} E(i, j-1) - \beta \\ V(i, j-1) - \alpha \end{cases}$$

$$F(i, j) = \max \begin{cases} F(i-1, j) - \beta \\ V(i-1, j) - \alpha \end{cases}$$

$$V(i, j) = \max \begin{cases} V(i-1, j-1) + \sigma(S_i, T_j) \\ E(i, j) \\ F(i, j) \\ 0 \end{cases}$$

PE design



$$E(i, j) = \max \begin{cases} E(i, j-1) - \beta \\ V(i, j-1) - \alpha \end{cases}$$

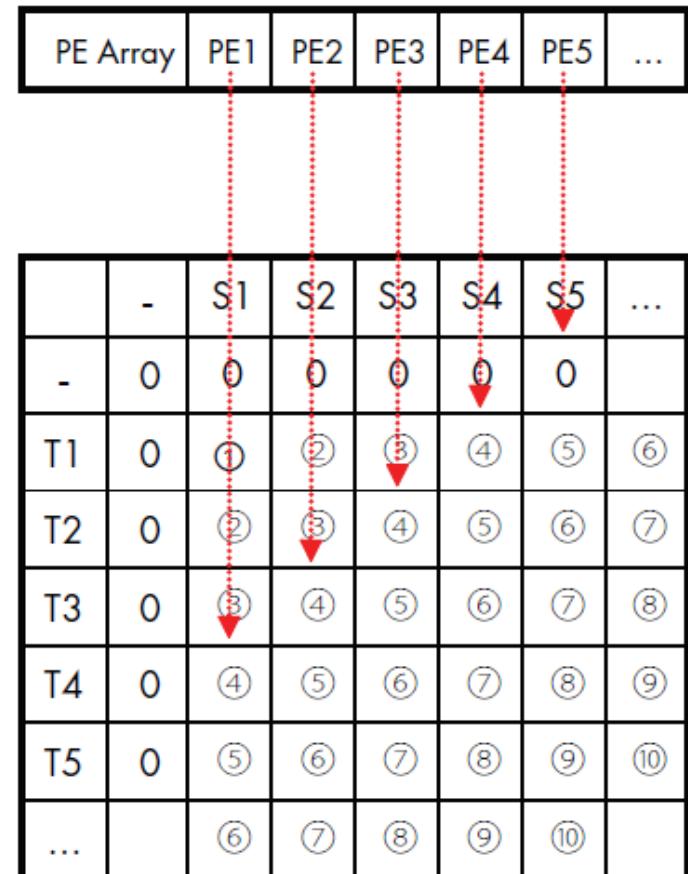
$$F(i, j) = \max \begin{cases} F(i-1, j) - \beta \\ V(i-1, j) - \alpha \end{cases}$$

$$V(i, j) = \max \begin{cases} V(i-1, j-1) + \sigma(S_i, T_j) \\ E(i, j) \\ F(i, j) \\ 0 \end{cases}$$

Parallelization

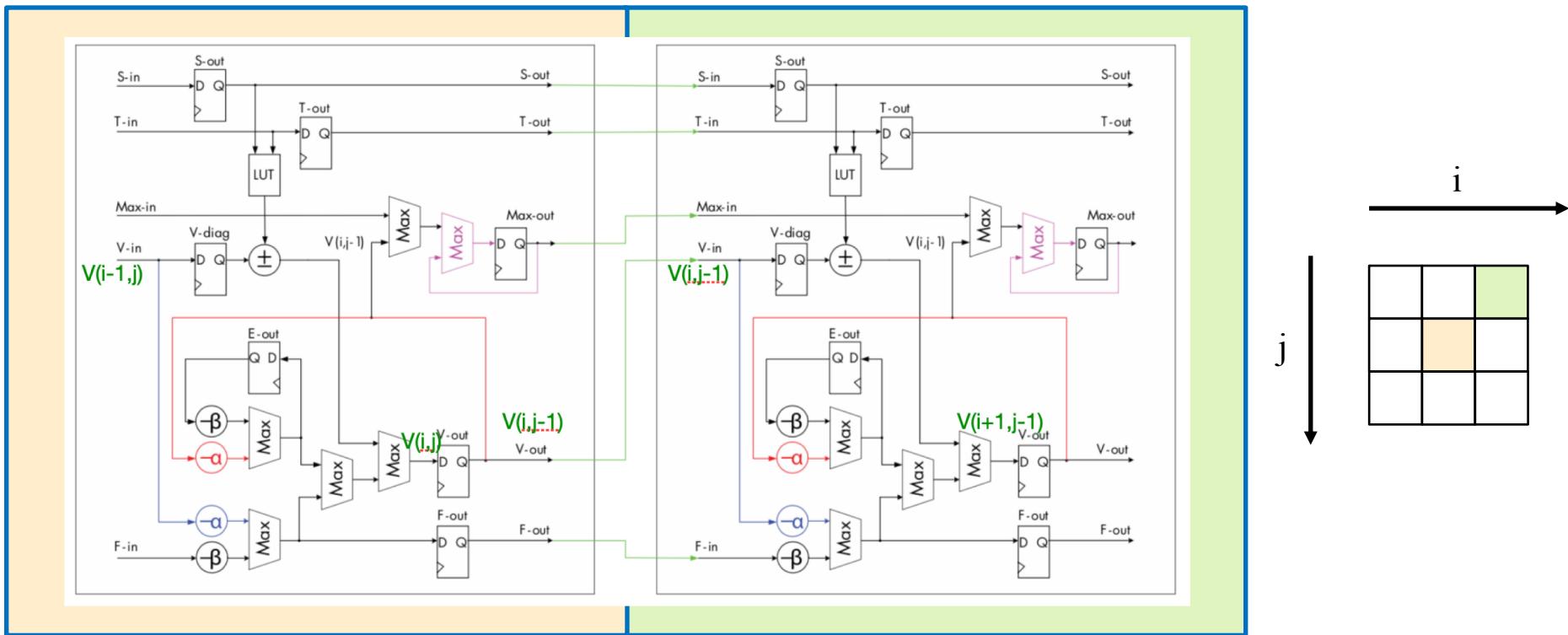
- Every node can be assigned value when it's upper, left, upper left nodes have value
- Which implies that it has a very good potential parallelity

	-	S1	S2	S3	S4	S5	...
-	0	0	0	0	0	0	
T1	0	①	②	③	④	⑤	⑥
T2	0	②	③	④	⑤	⑥	⑦
T3	0	③	④	⑤	⑥	⑦	⑧
T4	0	④	⑤	⑥	⑦	⑧	⑨
T5	0	⑤	⑥	⑦	⑧	⑨	⑩
...		⑥	⑦	⑧	⑨	⑩	



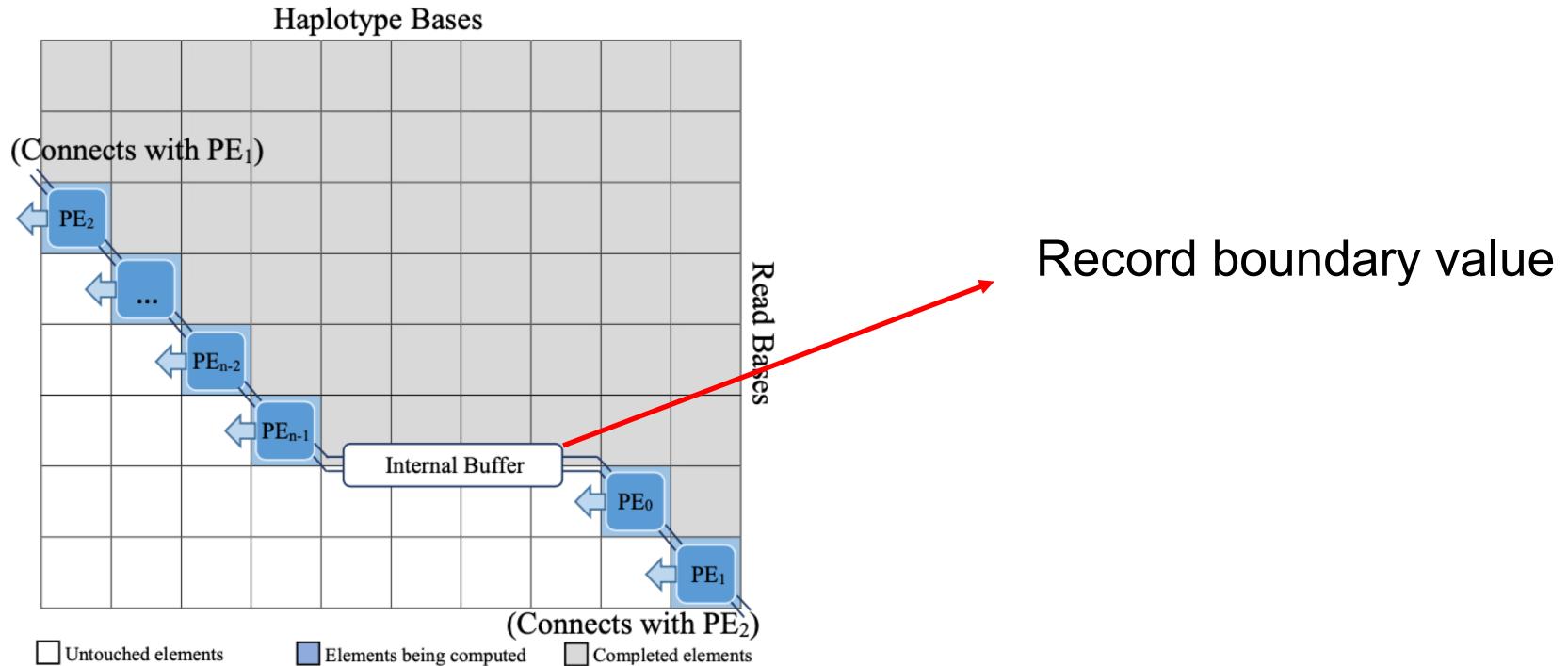
Architecture

- Example for combining PE :



Architecture

- When number of PEs < len(sequence)



- Length of sequence S = 256
- Length of sequence T = 256
- Number of PEs = 128
- Scoring Scheme :
 - Match = +8
 - Mismatch = -5
 - Gap_open(α) = -7
 - Gap_ext(β) = -3

- Input : Sequence S & Sequence T
- Output : Max alignment score

Some Idea

- How many bits to store sequence ?
- Max bit number of the score matrix ?
- How to store the boundary ?

Example

- Length of sequence S = 10
 - ACCTTAGGCA
- Length of sequence T = 10
 - GCCGGTTGCT

Golden

E		G	C	C	G	G	T	T	G	C	T
		-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞
A	-	-7	-7	-7	-7	-7	-7	-7	-7	-7	-7
C	-	-7	-7	-7	-7	-7	-7	-7	-7	-7	-7
C	-	-7	1	1	-6	-7	-7	-7	-7	1	-6
T	-	-7	1	9	2	-1	-4	-7	-7	1	-4
T	-	-7	-2	6	4	-3	7	4	-3	-2	9
A	-	-7	-5	3	1	-1	5	15	8	5	6
G	-	-7	-7	0	-2	-4	2	12	10	3	3
G	-	1	-6	-3	4	2	-1	9	16	9	6
C	-	1	-4	-6	1	12	5	6	13	11	4
A	-	-2	9	4	-1	9	7	3	10	21	14

Golden

F		G	C	C	G	G	T	T	G	C	T
		-	-	-	-	-	-	-	-	-	-
A	-∞	-7	-7	-7	-7	-7	-7	-7	-7	-7	-7
C	-∞	-7	-7	1	1	-2	-5	-7	-7	-7	1
C	-∞	-7	-7	1	9	6	3	0	-3	-6	1
T	-∞	-7	-7	-6	2	4	1	7	4	1	-2
T	-∞	-7	-7	-7	-1	-3	-1	5	15	12	9
A	-∞	-7	-7	-7	-4	-6	-7	-2	8	10	7
G	-∞	-7	1	-2	-5	4	2	-1	5	16	13
G	-∞	-7	1	-2	-5	1	12	9	6	13	11
C	-∞	-7	-6	9	6	3	5	7	4	6	21
A	-∞	-7	-7	2	4	1	2	0	2	3	14

Golden

V		G	C	C	G	G	T	T	G	C	T
		0	0	0	0	0	0	0	0	0	0
A	0	0	0	0	0	0	0	0	0	0	0
C	0	0	8	8	1	0	0	0	0	8	1
C	0	0	8	16	9	6	3	0	0	8	3
T	0	0	1	9	11	4	14	11	4	1	16
T	0	0	0	6	4	6	12	22	15	12	9
A	0	0	0	3	1	0	5	15	17	10	7
G	0	8	1	0	11	9	2	12	23	16	13
G	0	8	3	0	8	19	12	9	20	18	11
C	0	1	16	11	6	12	14	7	13	28	21
A	0	0	9	11	6	9	7	9	10	21	23

Reference

- Kun-Mao Chao, Algorithms for Biological Sequence Analysis class note, the Department of Computer Science and Information Engineering, National Taiwan University (NTU), Taipei, Taiwan, 2020.
- Zhang, Peiheng, Guangming Tan, and Guang R. Gao. "Implementation of the Smith-Waterman algorithm on a reconfigurable supercomputing platform." *Proceedings of the 1st international workshop on High-performance reconfigurable computing technology and applications: held in conjunction with SC07.* 2007.