



CS 68: BIOINFORMATICS

Prof. Sara Mathieson
Swarthmore College
Spring 2018



Outline: Feb 7

- Recap recursive traversal for DBG
- Recap global sequence alignment (Needleman-Wunsch)
- Local sequence alignment (Smith-Waterman)
- Alignment variations

Notes:

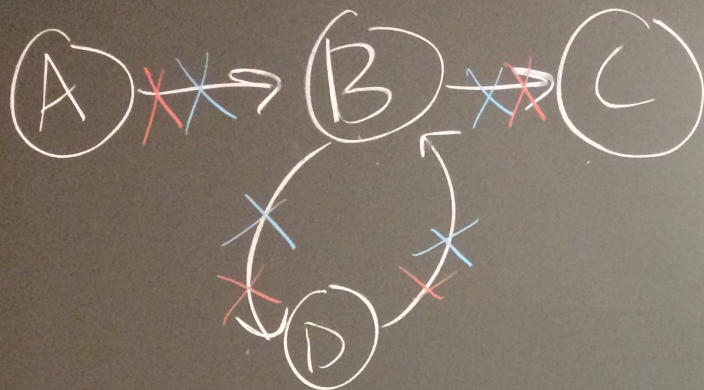
- Office hours today 1-3pm (in LAB)
- Partners for Lab 3 by tomorrow (Thurs at 11am)

Recap recursive traversal algorithm

```
find_path(u, path):  
    for each edge e = (u,v):  
        remove e  
        find_path(v, path)  
    append u to path
```

To use in python (using a list as a stack):

- Start with **path** as an empty list
- Start at **u** where $(\text{outdegree} - \text{indegree}) = 1$
- After completion, pop elements off **path** to find the correct order



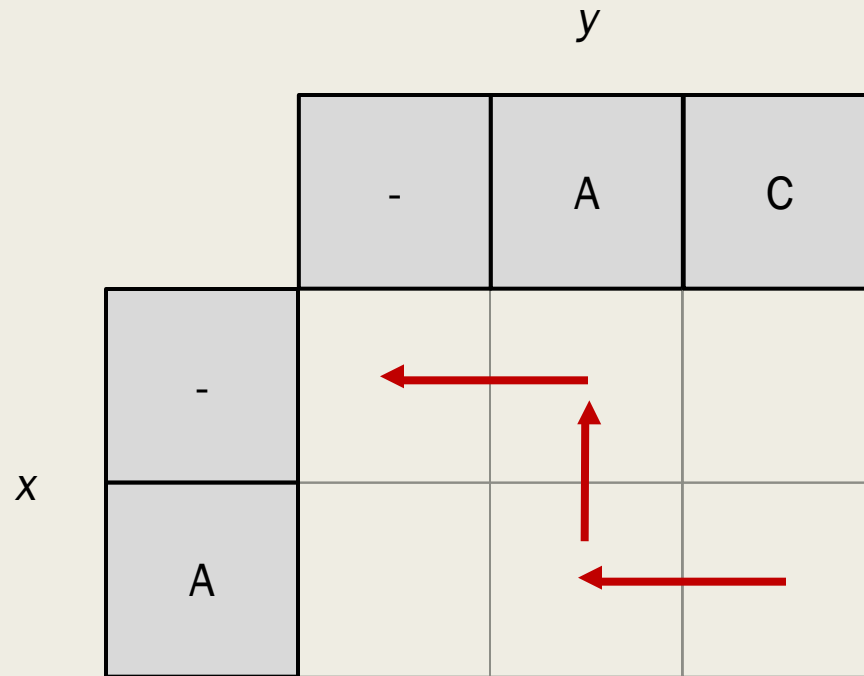
~~fp(C)~~
~~fp(B)~~
~~fp(D)~~
~~fp(B)~~
~~fp(A)~~
 function
 Stack 1

~~fp(D)~~ ← ~~fp(B)~~
~~fp(C)~~
~~fp(B)~~
~~fp(A)~~
 function
 Stack 2

path₁ = [C, B, D, B, A]

← path₂ = [C, B, D, B, A]

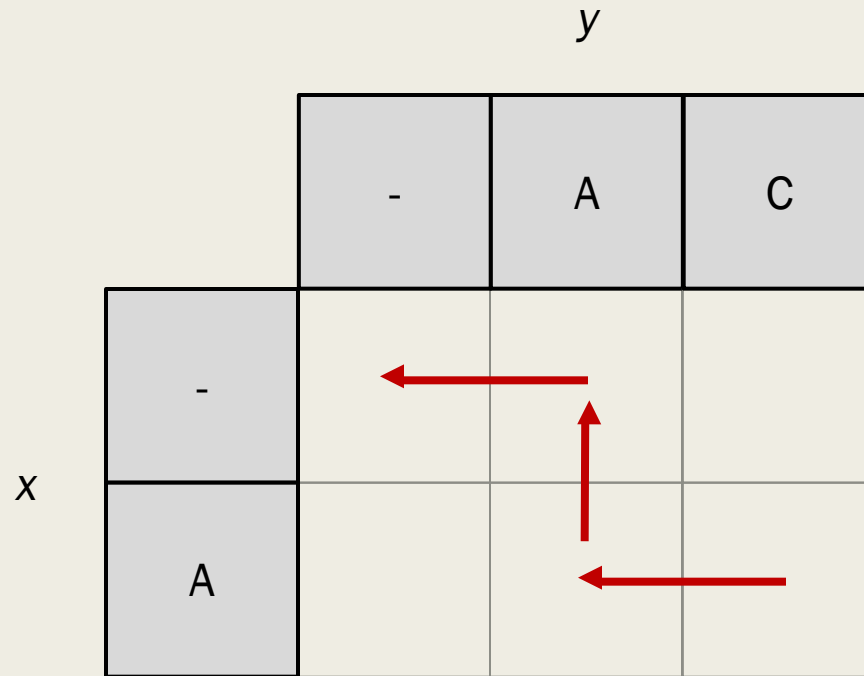
Recap quiz (discuss with a partner)



What alignment does the given trace-back represent?

When you finish, continue to work on the handout from last time!

Recap quiz (discuss with a partner)



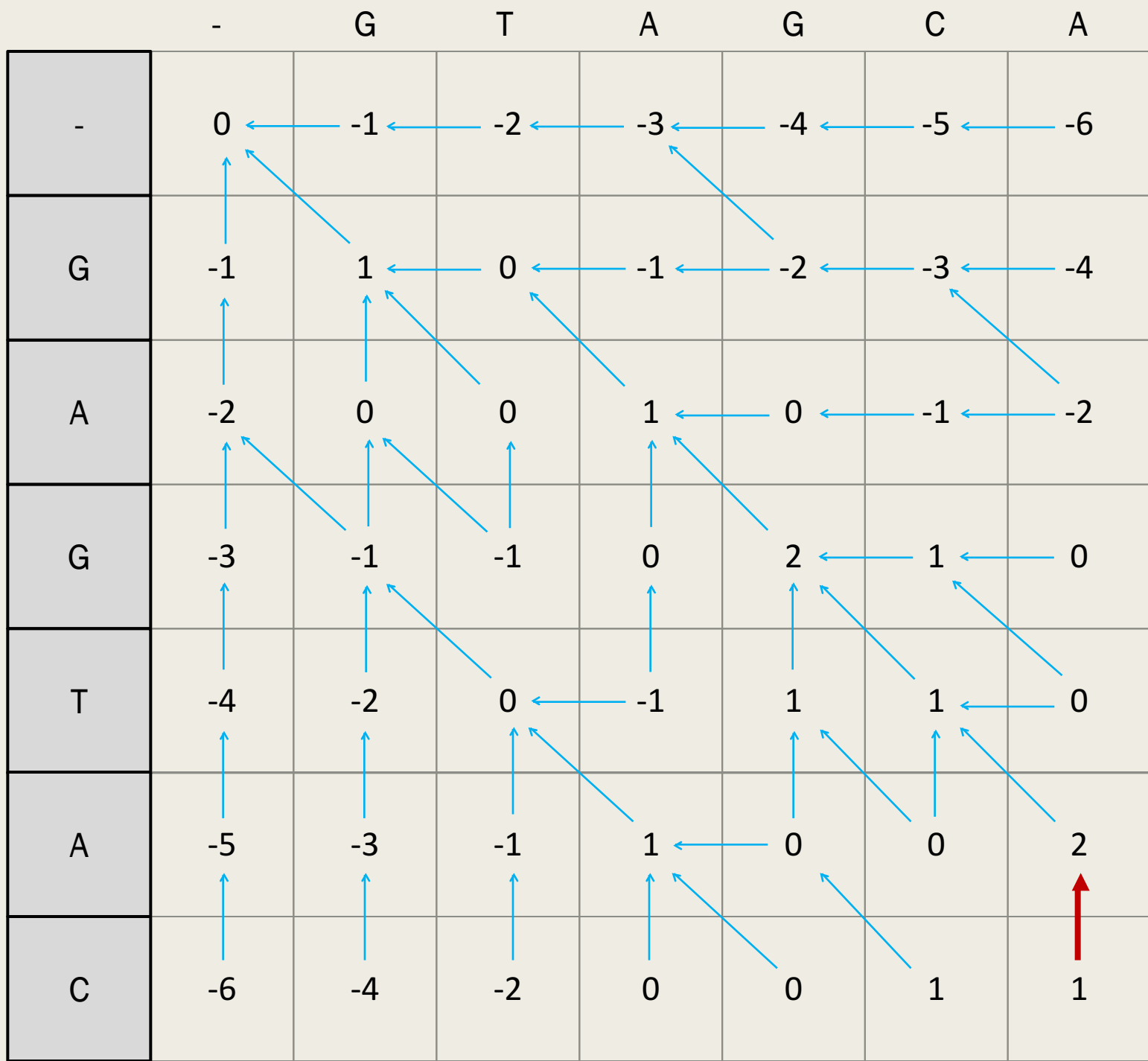
What alignment does the given trace-back represent?

$x: -A-$
 $y: A-C$

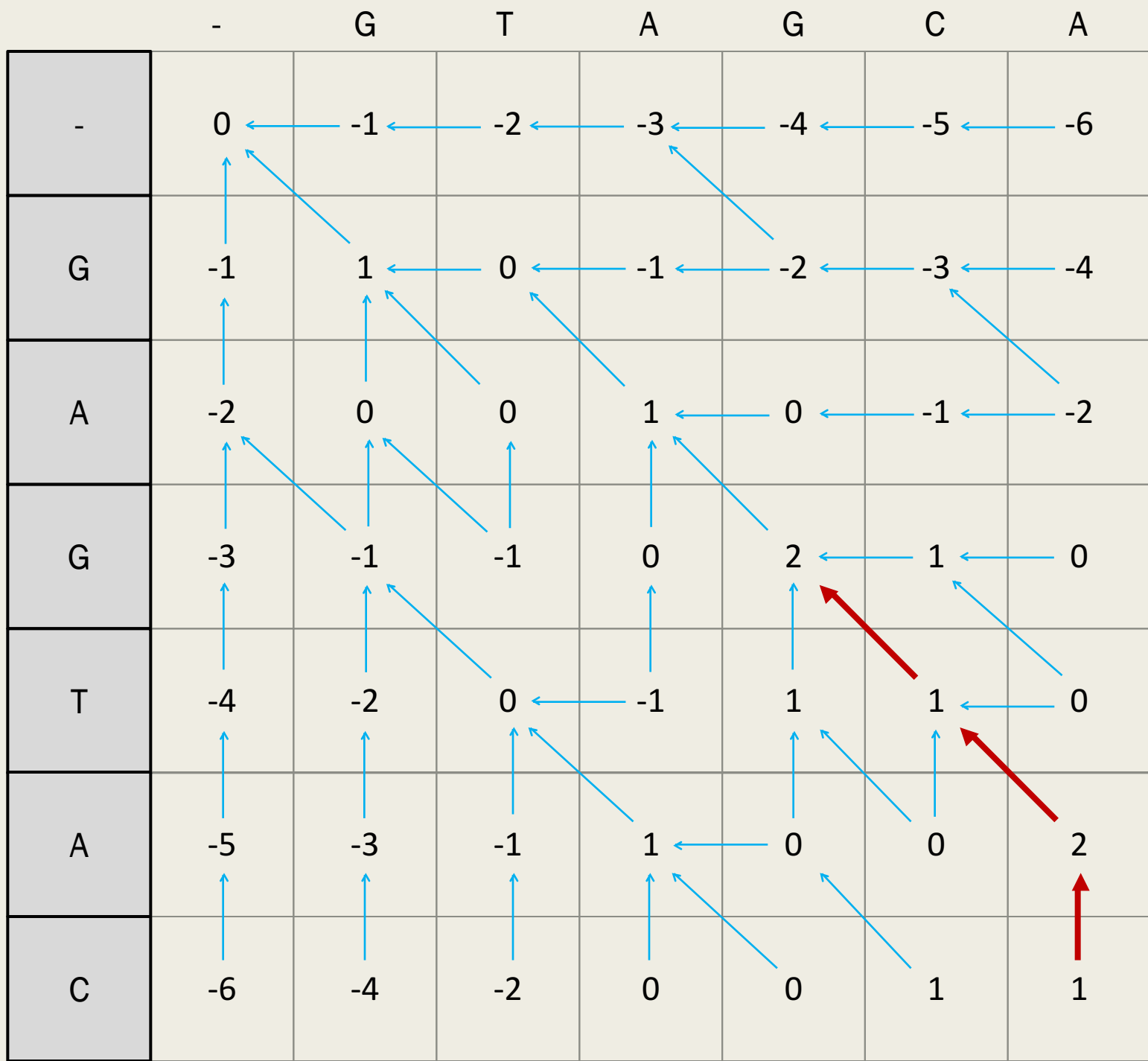
When you finish, continue to work on the handout from last time!

	-	G	T	A	G	C	A
-	0	-1	-2	-3	-4	-5	-6
G	-1	1	0	-1	-2	-3	-4
A	-2	0	0	1	0	-1	-2
G	-3	-1	-1	0	2	1	0
T	-4	-2	0	-1	1	1	0
A	-5	-3	-1	1	0	0	2
C	-6	-4	-2	0	0	1	1

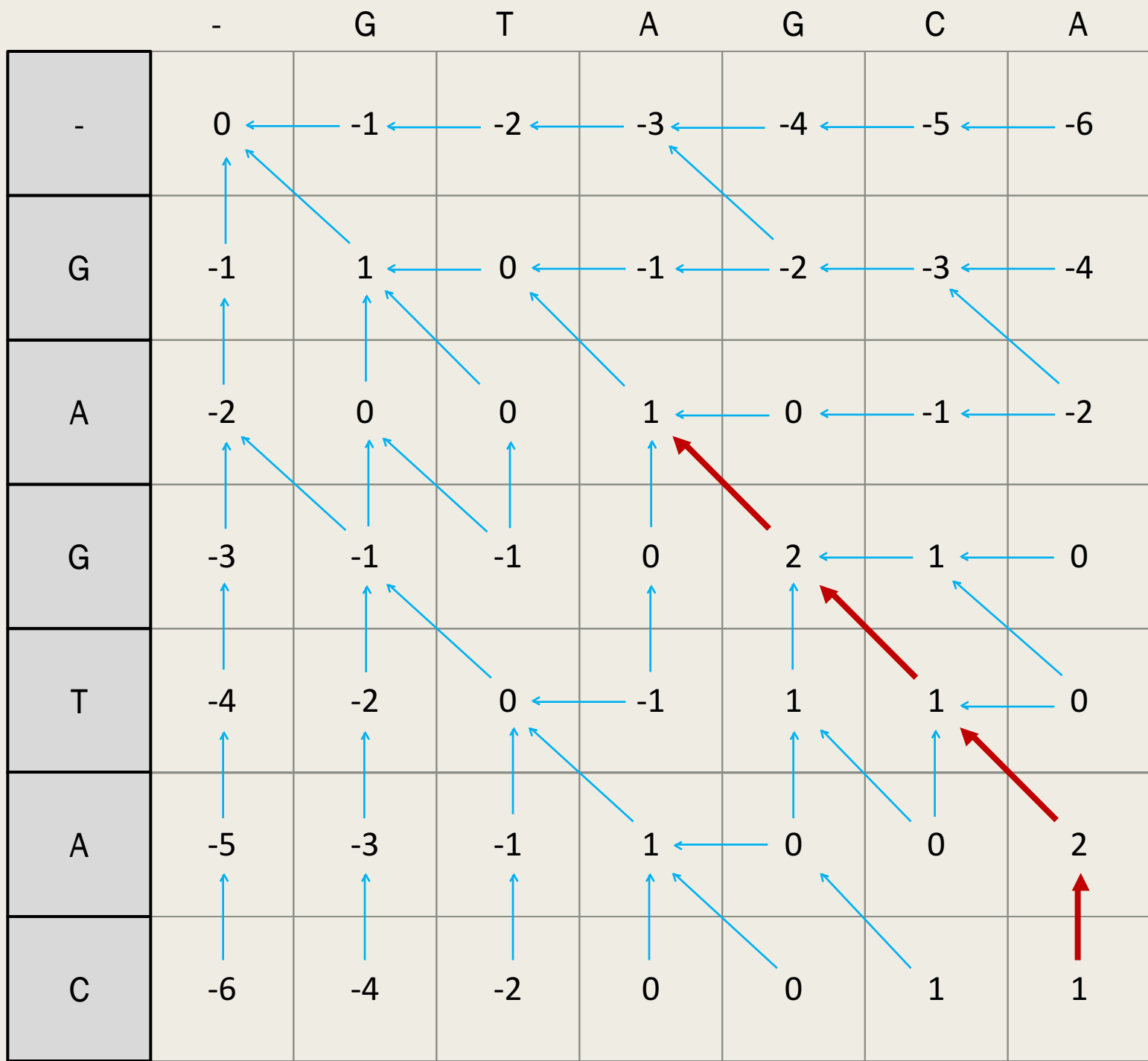
	-	G	T	A	G	C	A
-	0	-1	-2	-3	-4	-5	-6
G	-1	1	0	-1	-2	-3	-4
A	-2	0	0	1	0	-1	-2
G	-3	-1	-1	0	2	1	0
T	-4	-2	0	-1	1	1	0
A	-5	-3	-1	1	0	0	2
C	-6	-4	-2	0	0	1	1



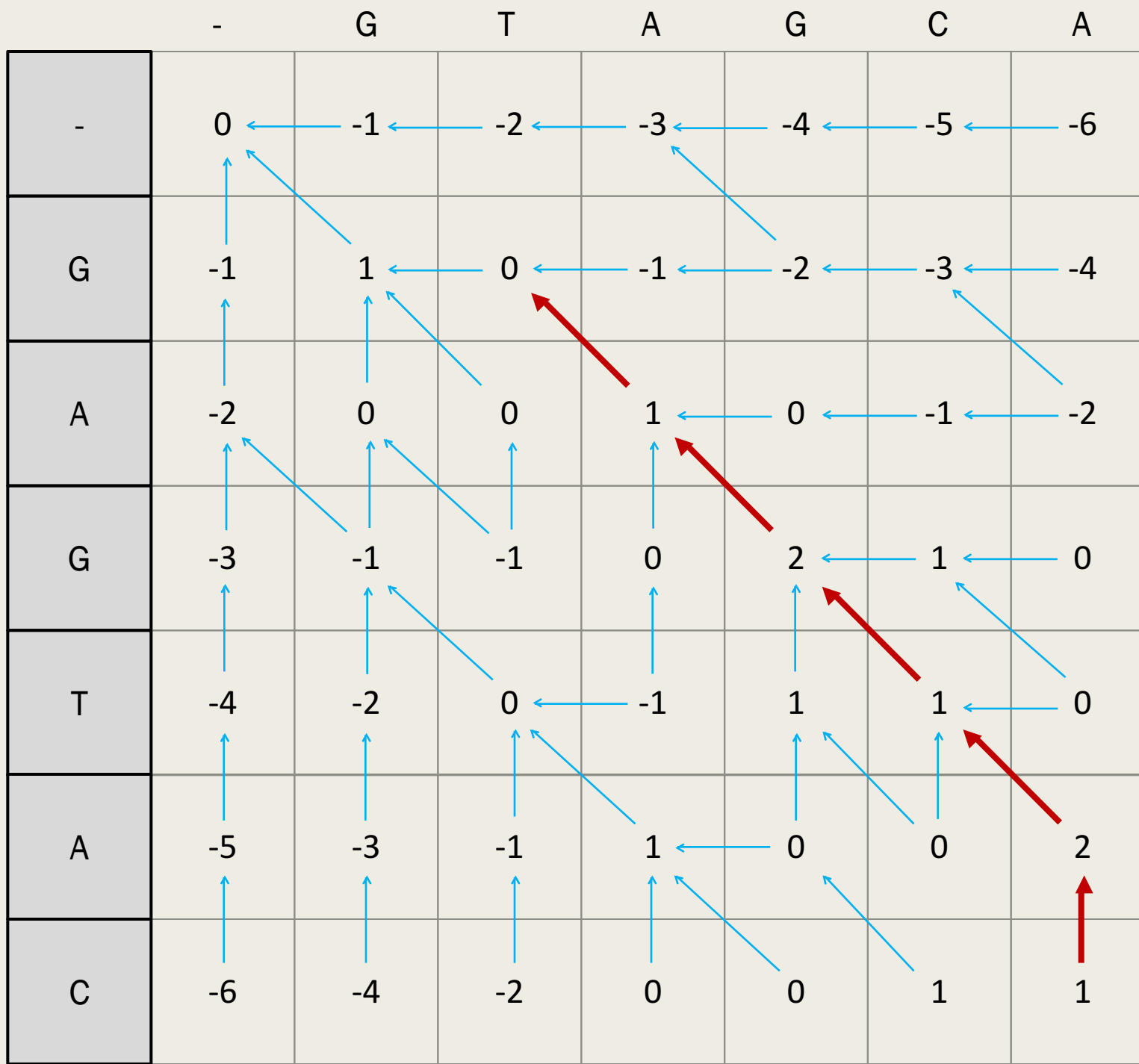
C
-

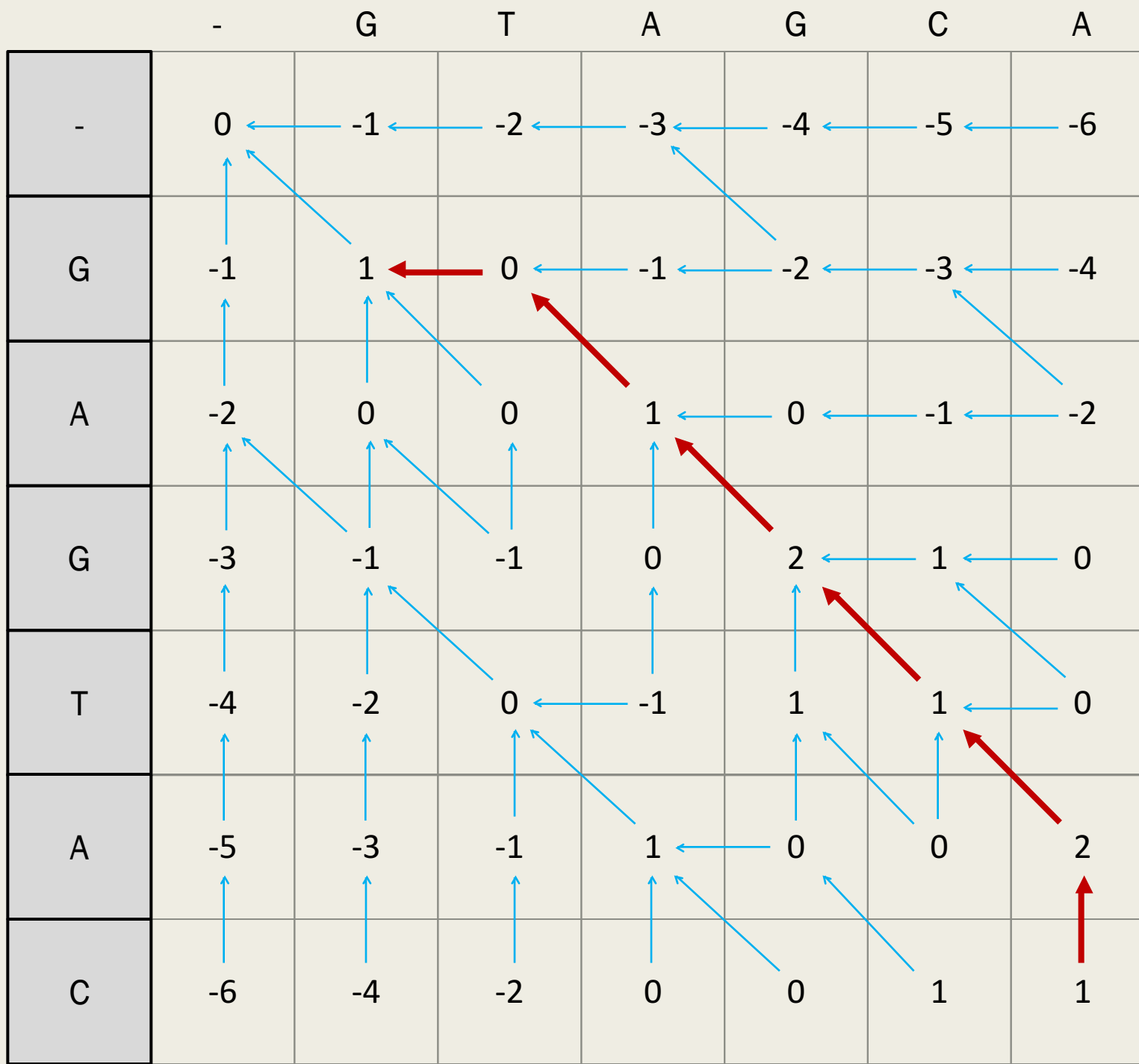


TAC
CA-

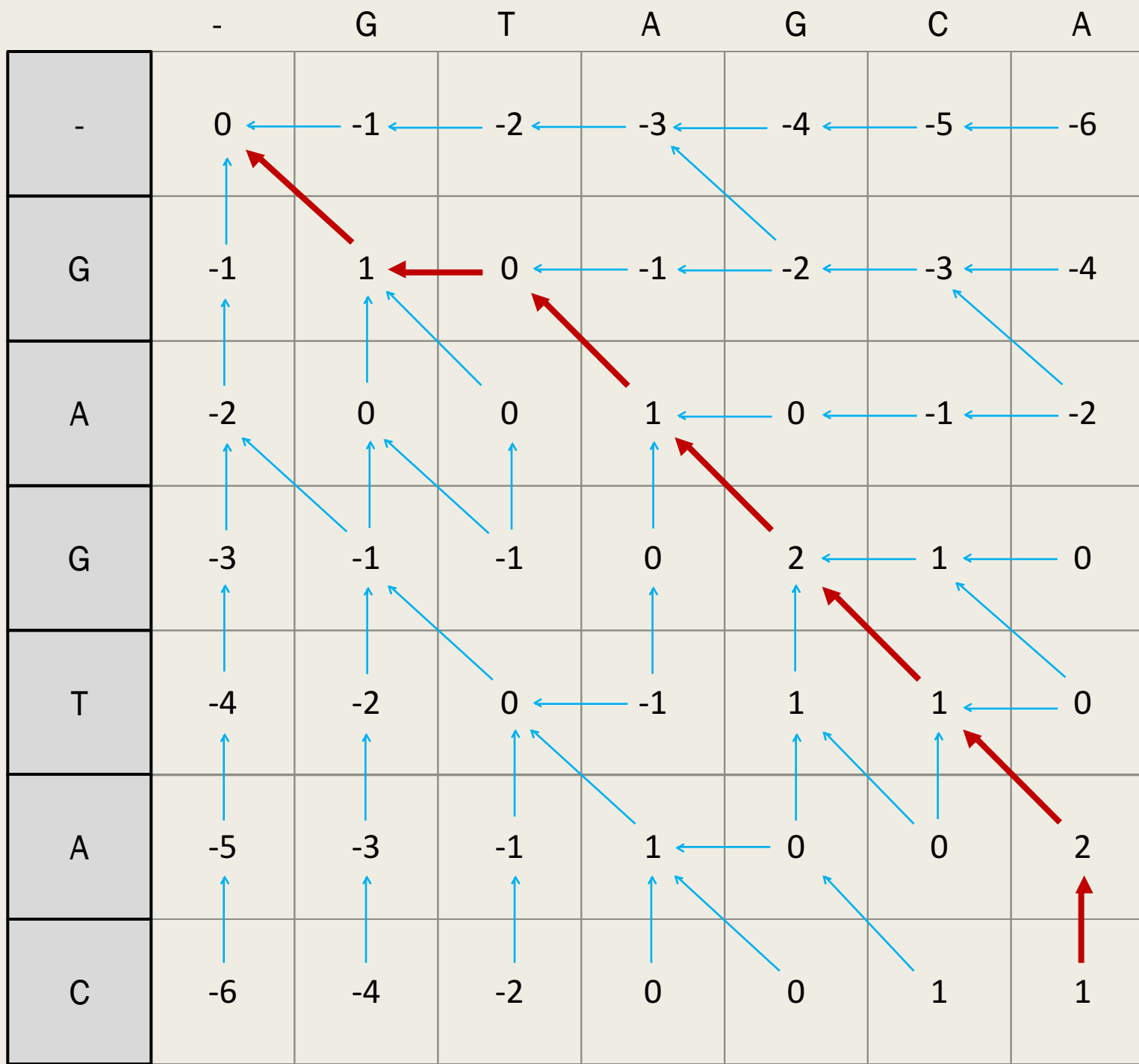


GTAC
GCA-





-AGTAC
TAGCA-



G-AGTAC
GTAGCA-

Protein alignment scoring matrix

Moving to proteins: BLOSUM match/mismatch matrix

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

Substitution scores between amino acids

$\left. \begin{matrix} ACA \\ ACA \end{matrix} \right\} S(x,y) = 3$

$\left. \begin{matrix} G-A \\ TTA \end{matrix} \right\} S(x,y) = -1$

runtime

$O(nm)$

if $n \approx m \rightarrow O(n^2)$

DNA: synonymous: does not change amino acid

$\rightarrow \left. \begin{matrix} TCT \rightarrow S \text{ a.a.} \\ TCC \rightarrow S \text{ a.a.} \end{matrix} \right\}$

$\left. \begin{matrix} TCT \rightarrow S \text{ a.a.} \\ GCT \rightarrow A \text{ a.a.} \end{matrix} \right\}$
non-synonymous

BLOSUM

	Q	I	V
Q	7	-3	-3
I	-3	5	4
V	-3	4	5

hydrophobic

$$m(a,b) \approx \log\left(\frac{p_{ab}}{q_a q_b}\right)$$

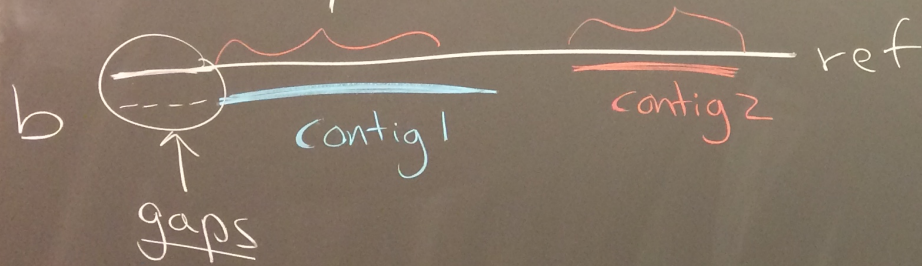
p_{ab} = prob of a change from $a \leftrightarrow b$

q_a = freq. of a in all proteins

Local alignment

Local Alignment

→ find the best alignments
between subsequences of
x & y



	A	T	C	C	G
C	0	0	0	0	0
G	0	0	0	1	0
A	0	0	0	0	2
T	0	1	0	0	1
C	0	0	1	3	2

ATC
ATC

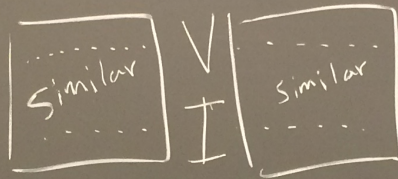
AA

base case: 0's in
the first row & column

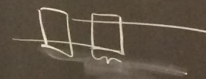
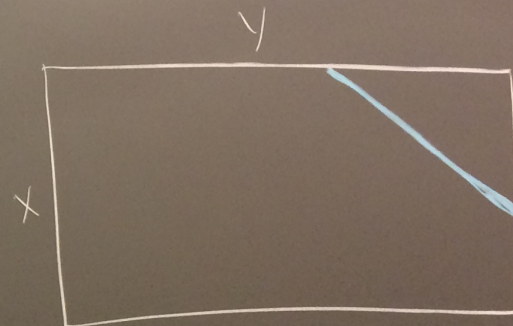
$$g = -1$$

recursion

$$S(i, j) = \max \begin{cases} 0 \\ S(i-1, j) + g \\ S(i, j-1) + g \\ S(i-1, j-1) + m(x_i, y_j) \end{cases}$$



traceback : Start from max
& trace back until
you hit zero



A