

Pawan Acharya

Proj_4

Report

3/17/22

1. Explain the time and space complexity of your algorithm by showing and summing up the complexity of each subsection of your code.

For unrestricted: The function `Array_pair` takes $O(1)$ time since we are just writing i,j to as a 2D array. The space complexity also seems to be $O(1)$, since we are not creating new space, rather overwriting.

However, the function `solve_sequence` takes $O(mn)$ since we are tracking through the backpoint and size of top and side sequence. The space complexity is $O(m+n)$.

Thus, the `full_align` (i.e. unrestricted) happens in $O(mn)$ time, where m is the size of the top sequence, and n is the size of the side sequence. Overall, the unrestricted algorithm has time and space complexity of $O(mn)$.

For Banded: The function, `banded-matrix` runs in constant time as we are just writing to a 2D array and not making any new space. So, the space complexity and time complexity both are $O(1)$.

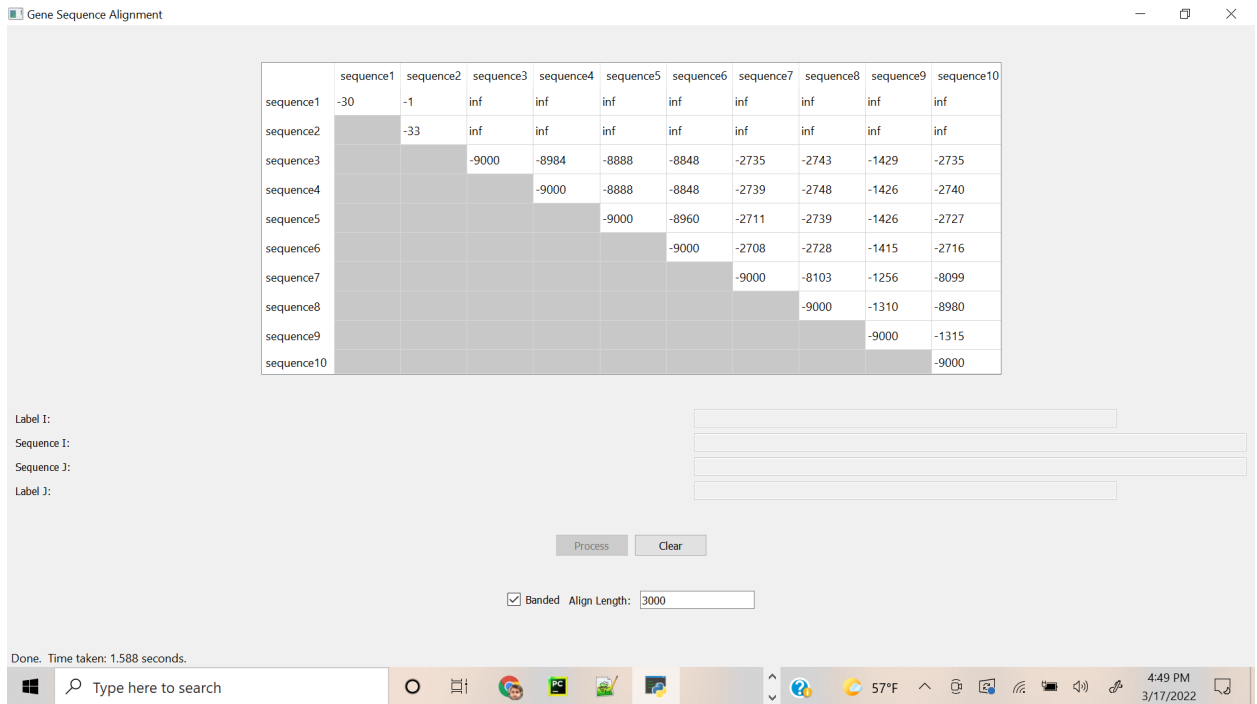
The function `bandedCompute_Seq` takes $O(kn)$ time, because it loops through backtrack using a while loop, in the columns and rows.

Overall, the time complexity of the banded algorithm is $O(kn)$, where k and n are the number of columns and rows.

2. Write a paragraph that explains how your alignment extraction algorithm works, including the backtrack.

The traceback array, as well as the alignment cost array, are updated throughout the algorithm with the proper value of where the `Array_pair` value came from. The letter "A" denotes the value from the cell uper to the current one, diagonal is indicated by a "D", and "L" indicates value from the left cell. When all of the costs have been determined, `Compute_Seq` is invoked. Then it adds the sequence letters to indices according to what letter is found, and continues until it finds cell (0,0) where "None" is found.

3. Results:



The extracted alignment for the first 100 characters of sequences #3 is

Gattgcgagcgatttgcgtgcgtgcat-ccc—gcttcactgatctcttggtagatcttttcataatctaaactttataaaaaacatccac
tcctgtgta-a

The extracted alignment for the first 100 characters of sequences #10 is

-a-taagagtgattggcggtccgtacgtaccctttctactctcaaaccttctgtagtttaaatc-taatctaaactttat--aaacggcactt
cctgtgt

	sequence1	sequence2	sequence3	sequence4	sequence5	sequence6	sequence7	sequence8	sequence9	sequence10
sequence1	-30	-1	456	456	456	456	456	456	456	456
sequence2		-33	448	448	448	448	448	448	448	448
sequence3			-300	-300	-288	-288	-126	-134	-79	-134
sequence4				-300	-288	-288	-126	-134	-79	-134
sequence5					-300	-300	-126	-134	-75	-134
sequence6						-300	-126	-134	-75	-134
sequence7							-300	-271	-154	-271
sequence8								-300	-183	-300
sequence9									-300	-183
sequence10										-300

Label 4:

gi|18033971|gb|AF391542.1|Bovine coronavirus isolate BCoV-LUN, complete genome.

Sequence 4:

jcgTgcgtgcaccccgcttcactgatctctgttagatcttttcataaactttataaaaaacatccactccctgtagtcta

Sequence 6:

jcgTgcgtgcaccccgcttcactgatctctgttagatcttttcataaactttataaaaaacatccactccctgtattcta

Label 6:

gi|17529670|gb|AF220295.1|Bovine coronavirus strain Quebec, complete genome.

Process

Clear

☐ Banded Align Length: 100

Done. Time taken: 0.426 seconds.

Type here to search



57°F

5:31 PM

3/17/2022