**Why do we need the dynamic programming algorithm for pairwise alignment?**

Suppose we wanted to align two sequences that were 100 and 90 amino acids long.

One way to get a global alignment would be to insert 10 gaps into the shorter sequence. In this case we can align each of the 10 gaps with any of the 100 characters in the longer sequence……but this has

100 choose 10 = 1.73 x 1013  (over 17 trillion) possible alignments!

We could also insert gaps into the longer sequence. Suppose the longer sequence had one gap, which could occur in 101 different positions (after each character, or before the first character). Then there would be

101 \* [100 choose 11] = 1.43 x 1016 (over 14 quadrillion) possible alignments…

And the longer sequence could have 2 gaps, 3 gaps, etc

Using the dynamic programming method, how many calculations are needed to align sequences that are length 100 and 90?

We need to fill in the dynamic programming matrix, which has (100 +1) rows and (90 + 1) columns. The number of calculations is on the order of 9,191.

Suppose a computer can do 1 million (1 x 106) calculations per second).

* Brute force approach: about 454 years (only considering the gaps above – in reality, it will take much, much, much longer)
* DP takes: 0.009 seconds

Dynamic Programming Example

***Note: For all alignments, we will use a linear gap penalty of 4 points, a match score of +5 points, and a mismatch score of -1 point.***

Complete the dynamic programming matrix below to find the optimal alignment score and optimal alignment for THEM and TEA.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | T | E | A |
|  | 0 | -4 | -8 | -12 |
| T | -4 |  |  |  |
| H | -8 |  |  |  |
| E | -12 |  |  |  |
| M | -16 |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Si,j = max{ |  |  |  |  |
|  |  |  |  |
|  |  |  |  |