CMSC 423: Sequence Alignment

Part 6

Running times

- All these algorithms run in O(mn) – quadratic time
- Note – this is significantly worse than exact matching

Running times

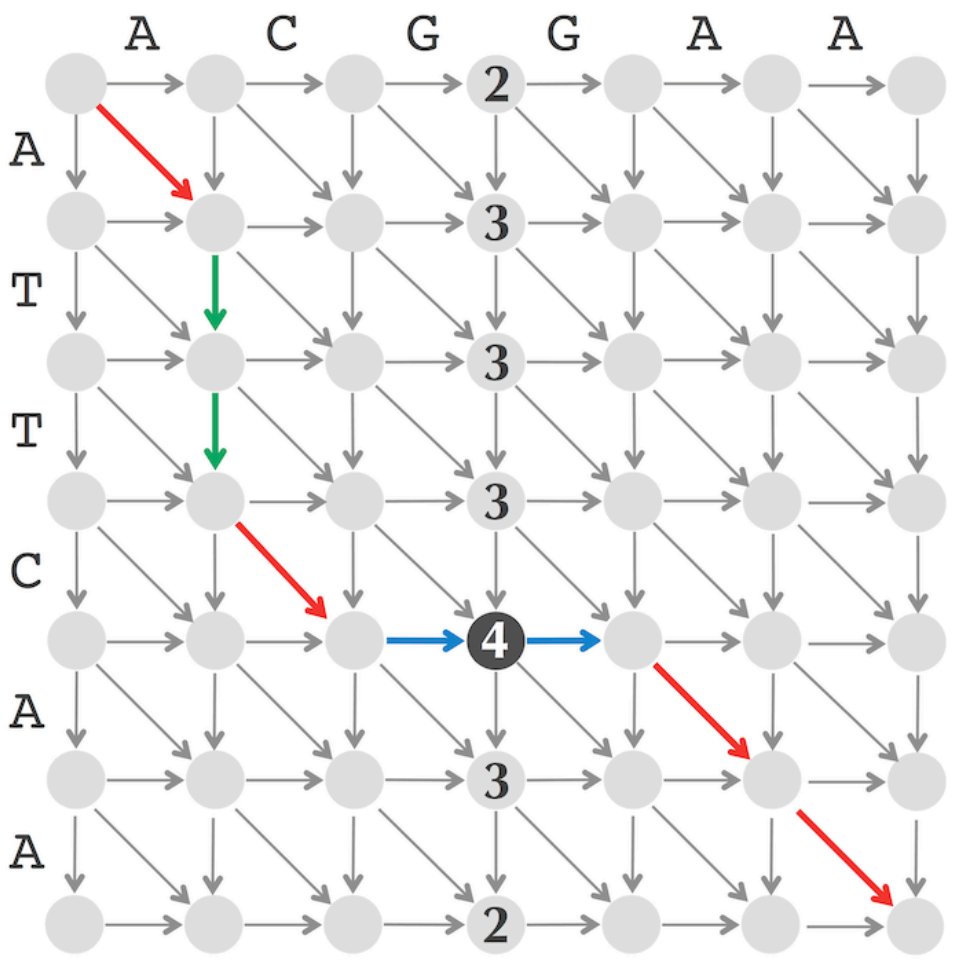
- All these algorithms run in O(mn) – quadratic time
- Note – this is significantly worse than exact matching
- BTW, how much space is needed?
  - If we only need to find the best score (not the exact alignment as well) – O(min(m,n))
  - If we the exact alignment too – O(m·n)

Running times

- All these algorithms run in O(mn) – quadratic time
- Note – this is significantly worse than exact matching
- Next week we'll talk about speed-up opportunities
- BTW, how much space is needed?
  - If we only need to find the best score (not the exact alignment as well) – O(min(m,n))
  - If we the exact alignment too – O(m·n)
  - If we need to find the best alignment – elegant divide and conquer algorithm leads to linear space solution

The Middle Node Problem

- Middle node = the node on the longest path belonging to the middle column



We can find a longest path's middle node without having to construct the path in the alignment graph

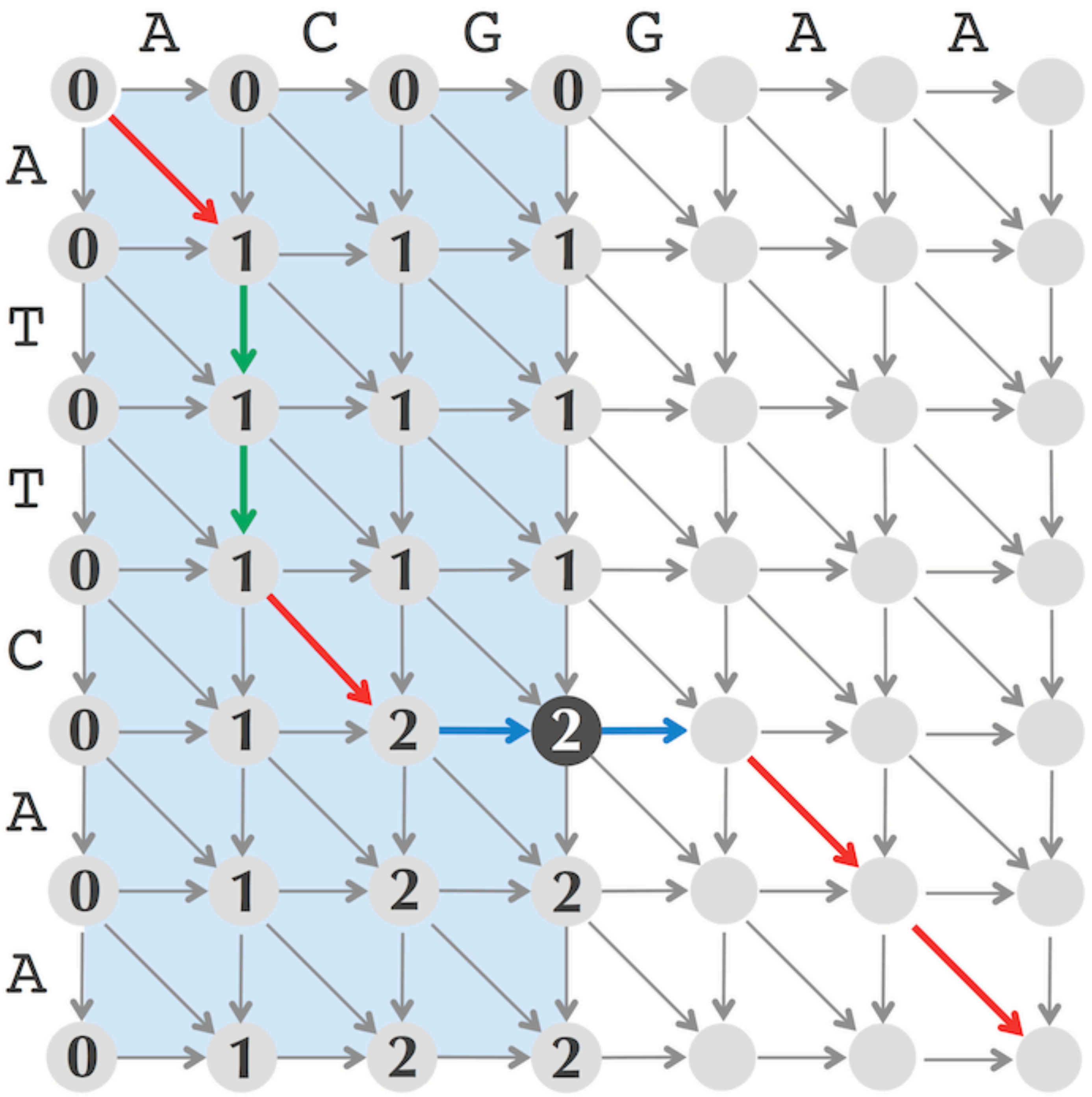
- *i*-path passes through the middle column at row *i*
- For each *i* between 0 and *n*, find the length of the longest *i*-path

$$\text{Length}(i) = \text{FromSource}(i) + \text{ToSink}(i)$$

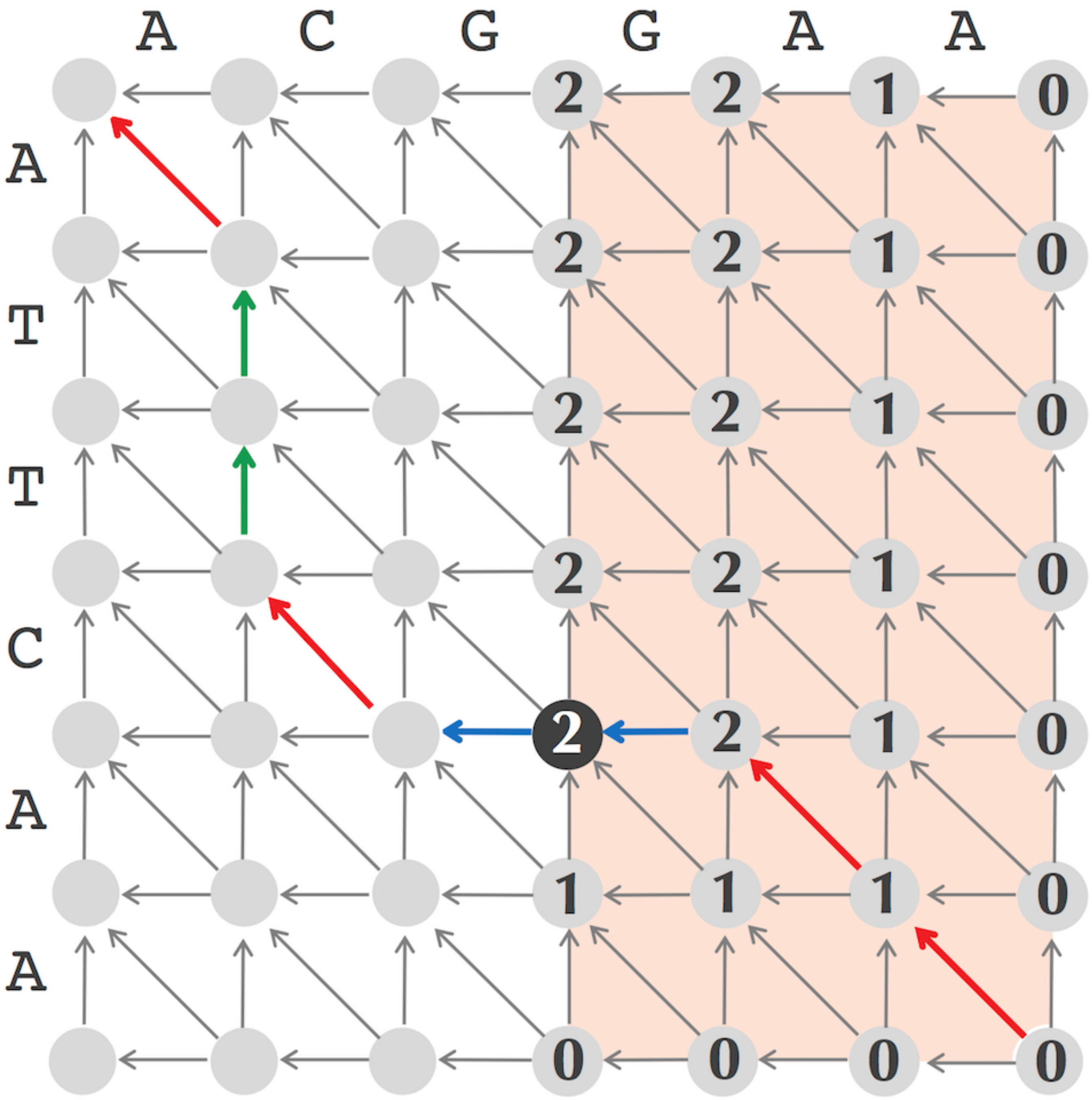
FromSource(*i*) = longest path from source to (*i*, *middle*)

ToSink(*i*) = longest path from (*i*, *middle*) to sink

FromSource(i) can be computed in $\mathcal{O}(n)$ space and $\mathcal{O}(n \cdot m/2)$ time



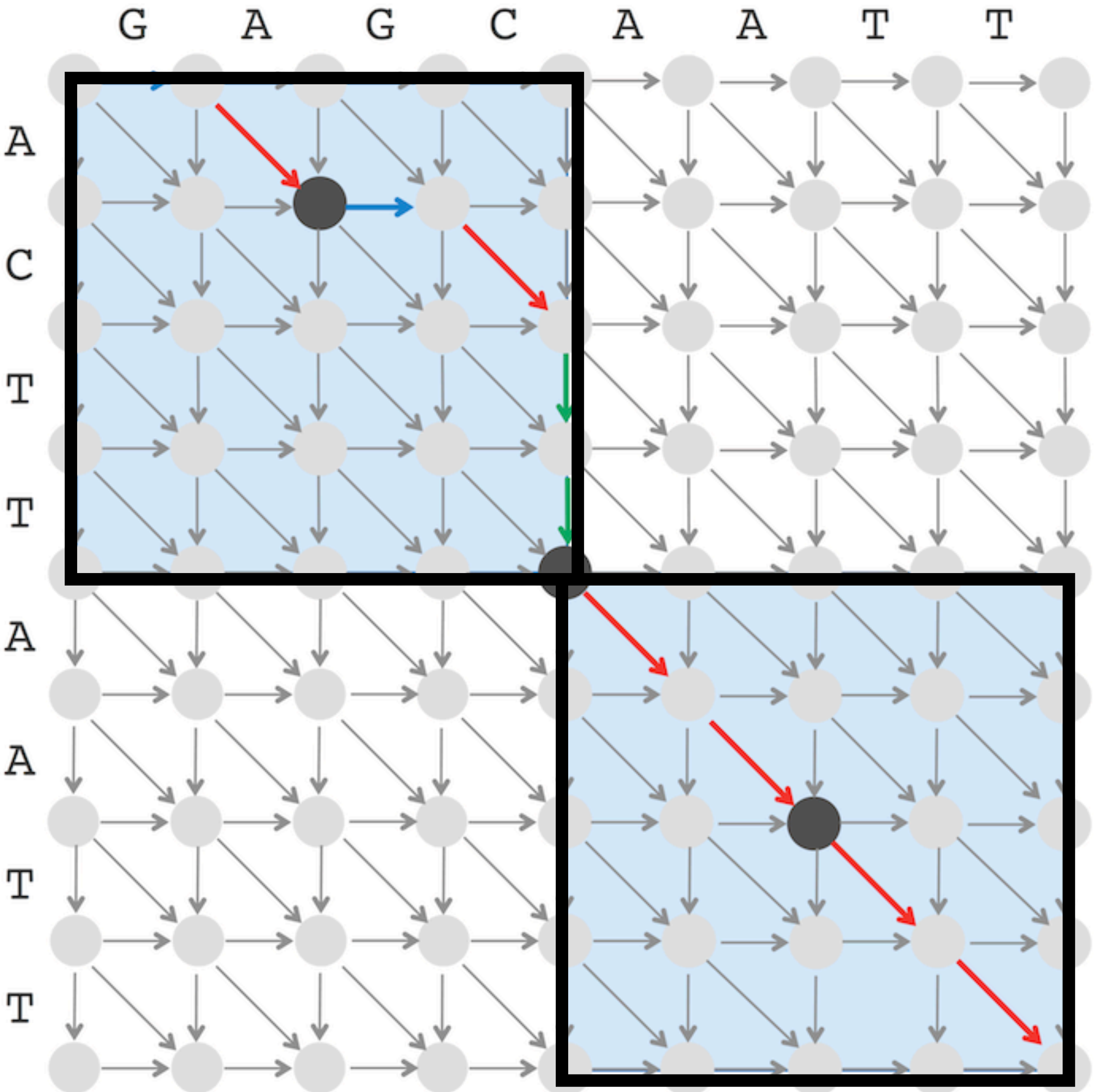
ToSink(i) can also be computed in $\mathcal{O}(n)$ space and $\mathcal{O}(n \cdot m/2)$ time

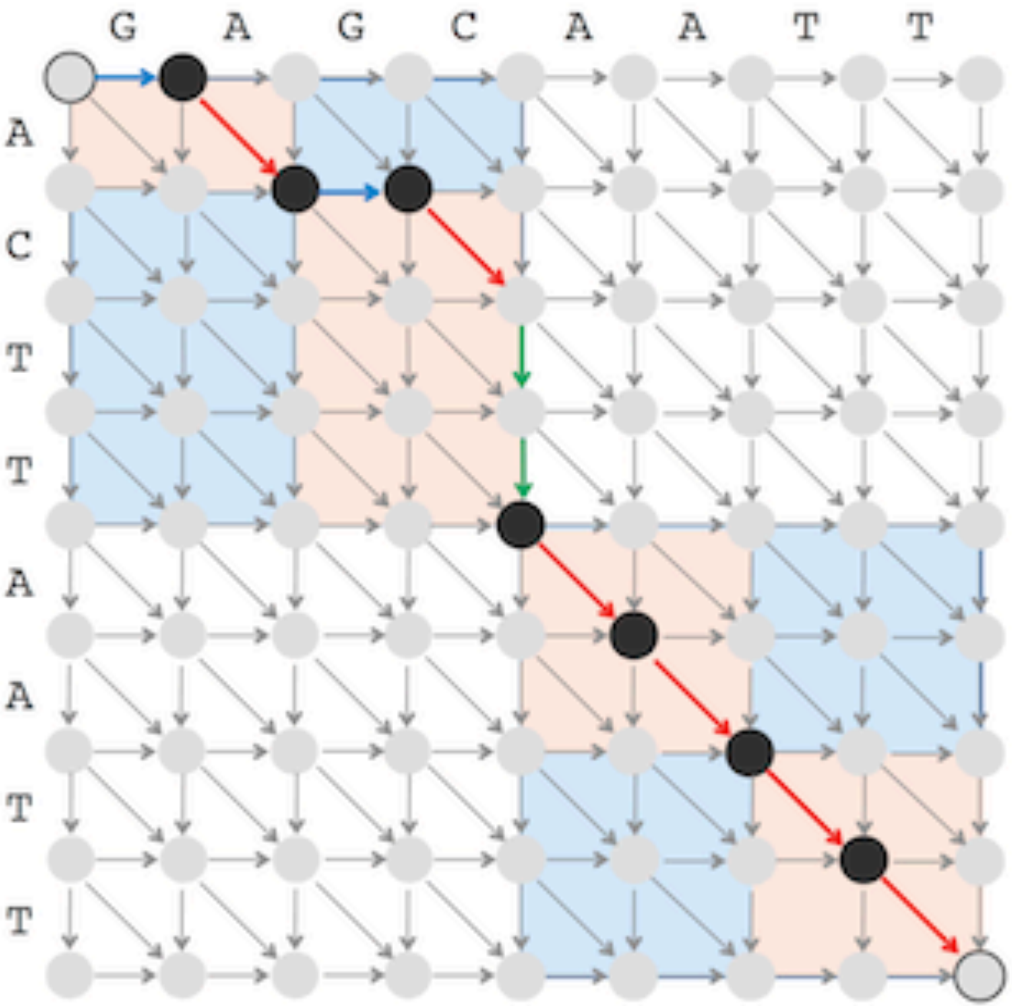


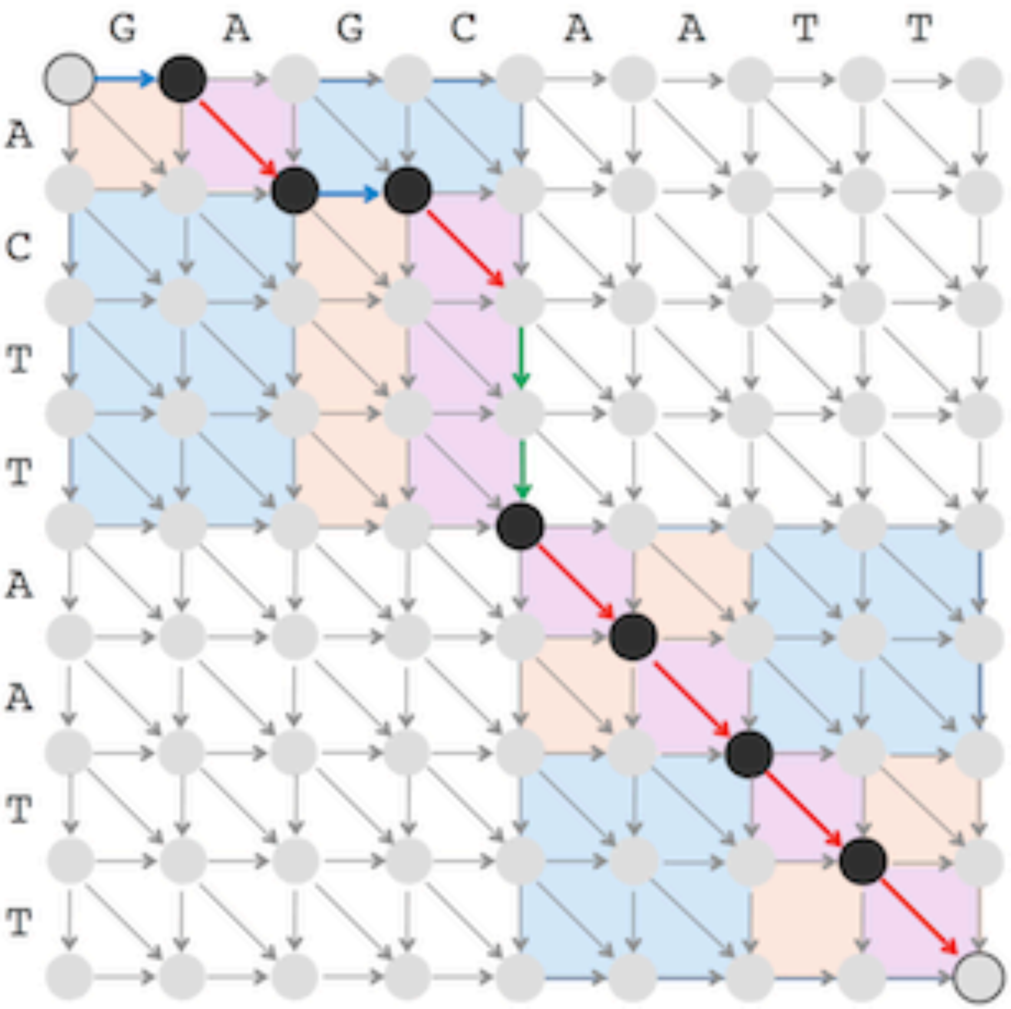
Length(i) = FromSource(i) + ToSink(i)

- Can be computed in linear space
- Runtime is proportional to $n \cdot m/2 + n \cdot m/2 = n \cdot m$

Now, we divide the problem of finding the longest path form (0,0) to (n,m) into two subproblems







$$n \cdot m + n \cdot m/2 + n \cdot m/4 + \cdots < 2 \cdot n \cdot m = O(n \cdot m)$$

The Middle Edge Problem