6.851 Advanced Data Structures (Spring'07)

Prof. Erik Demaine TA: Oren Weimann

Problem 5 *Due: Monday, Mar. 19*

Be sure to read the instructions on the assignments section of the class web page.

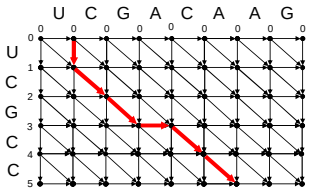
Approximate Pattern Matching Under the Edit Distance Metric.

Given a text T of length n and a pattern P of length m, the one-shot *approximate string matching problem* asks to find all occurrences of P in T with error k. In class we have seen how to solve this problem in $O(nk)$ time under the Hamming Distance metric. We now address the problem under the Edit Distance metric (i.e., the minimum number of character insertions, deletions, and mismatches needed for an exact match).

There is a simple dynamic program that solves the problem in $O(mn)$ time and space. Let $ed(i, j)$ be the minimum edit distance when matching the ith prefix of P with some substring of T that ends in $T[j]$. Then

$$ed(i, j) = \min\{ed(i-1, j) + 1, ed(i, j-1) + 1, ed(i-1, j-1) + C\}$$

where $C = 0$ if $P[i] = T[j]$ and $C = 1$ otherwise. The halting conditions are $ed(0, j) = 0$ and $ed(i, 0) = i$. The dynamic program table for $T = UCGACAAG$ and $P = UCGCC$ is



A horizontal edge corresponds to deleting the character (above) from T, a vertical edge corresponds to deleting the character (to the left) from P, and a diagonal edge corresponds to matching or mismatching these two characters. The red path in the figure thus shows that P occurs in T with $k = 3$ edits (ending at position 6 in T).

By filling in the entire dynamic program table we can solve the problem in $O(nm)$ time by simply looking at the last line of the table and outputting every cell $ed(|P|, j)$ with value less than k.

- Show how to solve the problem in $O(nk)$ time and space.