

Related Work :

Most of string matching algorithms are based on 3 main algorithms BF (naive) -KMP algorithm -BM algorithm, so we will discuss them in this section.

-BF algorithm (Naive) :

bf algorithm is known as the brute force algorithm or naive whose basic idea is simple, it compares the characters of the pattern with the text one by one, the pattern moves forward by one character and the algorithm starts a new iteration of comparison, so it tries all possible alignments.

it's easy to understand but on the other hand it's not efficient and has a large number of repeated comparisons, with a complexity = $O(mn)$

-KMP algorithm :

The naïve algorithm compares the character several times, in order to overcome this drawback , Knuth, Morris and Pratt proposed the KMP algorithm in 1977.

the main difference it has with the naive one that it processes the pattern before comparing it with the text which results in LPS table. The goal is to compare each character in the text only once.

The way it works in brief:

First we construct the LPS table according to the pattern then the characters of the text are read and compared one by one, the shift of the pattern is decided based on the states in LPS table.

The total comparisons is at most $2n$, twice the length of the text.

The complexity of the pre-processing procedure is $O(m^2)$, generally the consuming time of the pre-processing can be ignored but it increases the complexity of the algorithm's implementation.

-BM algorithm :

When BM algorithm was proposed it was a turning point in the history of string matching algorithms.

Before : the direction of character comparison was consistent to the movement direction of the pattern, i.e. from left to right which results that each character must be compared at least once and no character and be skipped.

After : the direction of character comparison is different from the movement direction of the pattern.

How? when the pattern is positioned over the text at a place, it compares the characters of the pattern with the

corresponding characters in the text one by one from the right of the pattern to the left side; if the matching is failed, then based on the comparison information of current iteration, it right shifts the pattern for one to several characters, based on two rules (BAD CHARACTER RULE, GOOD SUFFIX RULE) depends on which one offers bigger text overlap.

The most important feature of BM algorithm is that the characters comparison is from the right of P to the left. Its advantage is that many characters in text T may be ignored during the comparison, in this way the efficiency of algorithm could be substantially improved.

The complexity of algorithm BM is $O(m)$, In most cases of the practical application, the efficiency of BM algorithm is higher than of KMP algorithm.

After BM algorithm was proposed, there were some algorithms to improve it like (BMH, BMHS BMH2C)

And we will discuss that topic in the next section.