Time Complexities of Comb Sort

Best Case Complexity:

In the best case scenario, the array is already sorted. Even though Combsort still performs comparisons, it does not perform any swaps because the array is already in order. In this case, the time complexity is approximately O(n log n). This is because the gap sequence helps in reducing the number of comparisons, but it's not as efficient as some other sorting algorithms like quicksort or mergesort.

Worst Case Complexity:

The worst-case time complexity of Comb sort is O(n^2). This occurs when the array is in reverse order or nearly in reverse order, causing Combsort to perform a large number of comparisons and swaps. Despite its improvements over bubble sort, Combsort's worst-case time complexity is still quadratic.

Average Case Complexity:

The average case complexity of Comb sort is also O(n^2). This is because, on average, Combsort will perform a similar number of comparisons and swaps as it would in the worst-case scenario. The reduction in the gap sequence helps in most cases, but it does not significantly alter the average time complexity compared to the worst case.