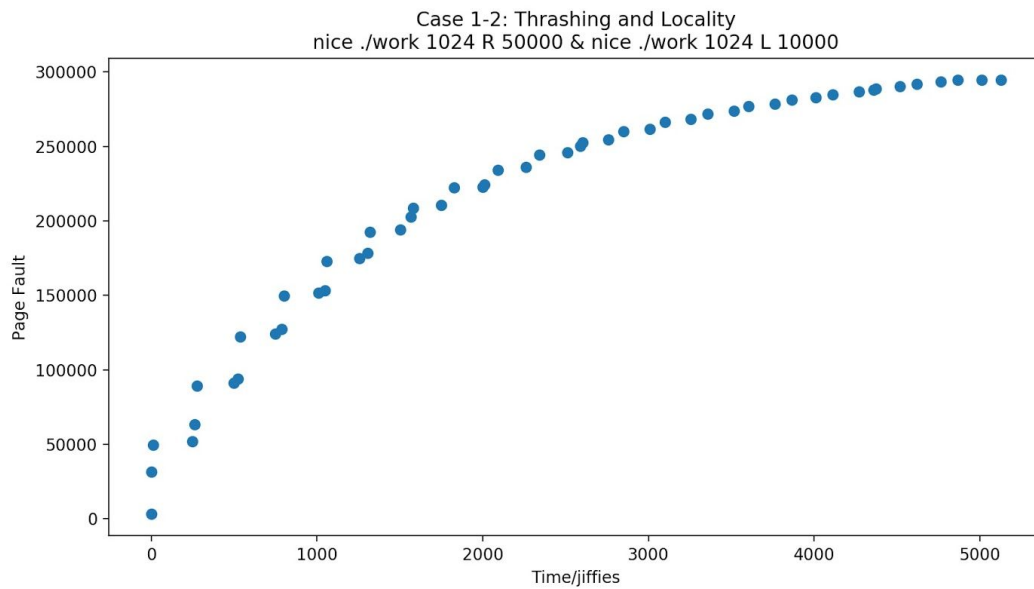
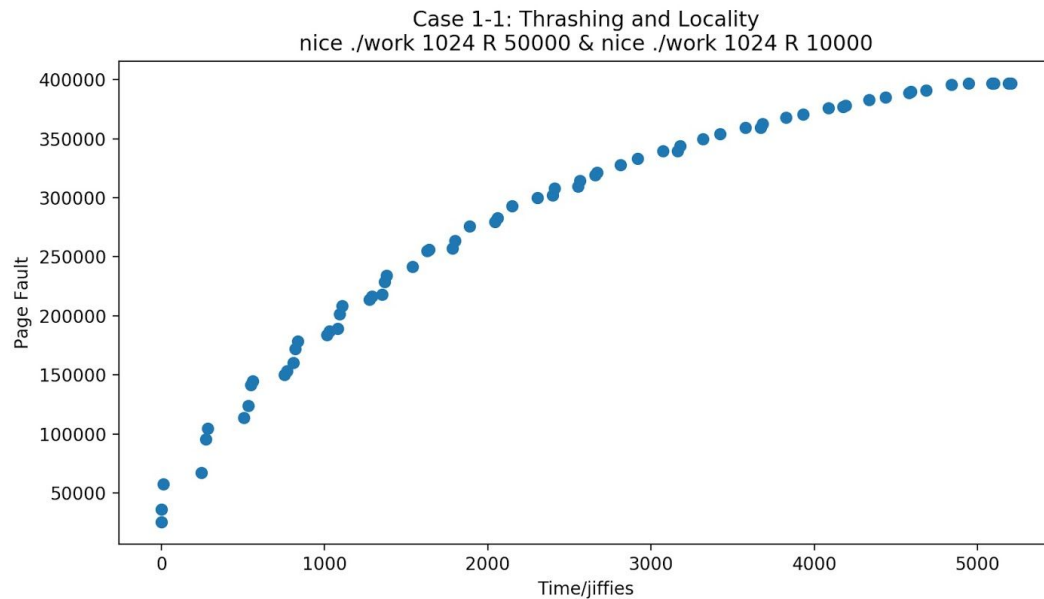


# MP3 Report

## Case 1



There are two sub-case in case 1:

Subcase 1: two process with random access

Subcase 2: one process with random access, one process with locality-based access

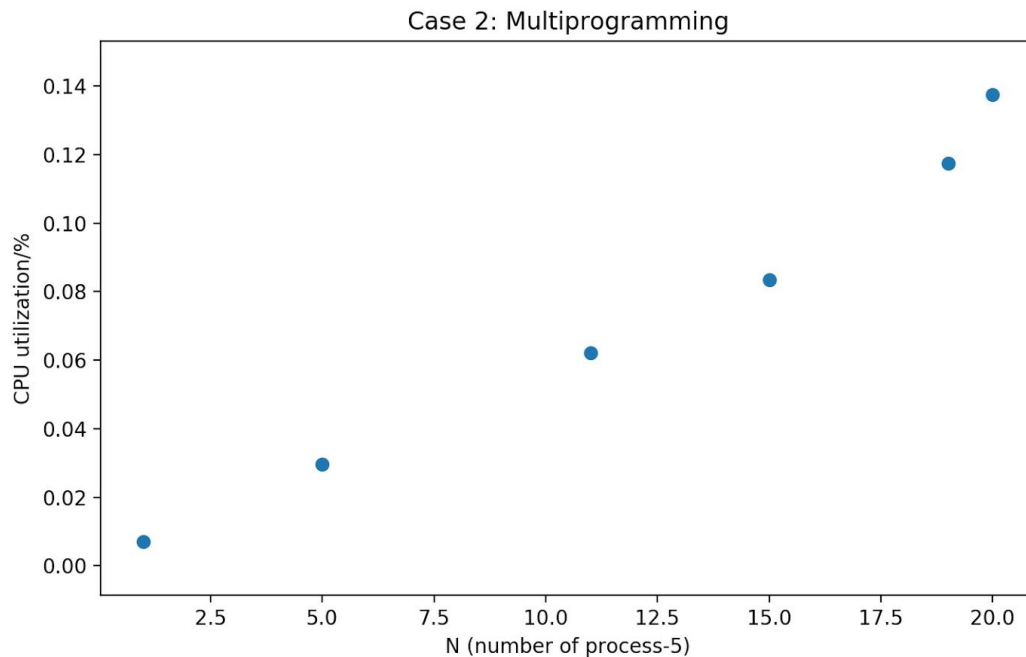
Be comparing the above two graphs, we can see that the overall trend is similar, the total page fault count increases over time.

But there are two differences between these two subcases:

1. The total page fault count is different. In subcase 1, the page fault count is almost 400000. While the page fault count is almost 300000 in subcase 2.
2. The runtime is slightly different. It is 5150 jiffies for subcase 1. And it is 5123 jiffies for subcase 2.

The major reason for these differences between the two subcases is that there is one process with locality-based access in the subcase 2. And the locality-based access is more aligned with the memory management system which is tuned for spatial and temporal locality access patterns. So the locality-based access can reduce page fault, leading to better performance.

## Case 2



As shown in the graph, the CPU utilization increase over the N (the number of process 5). A maximum is expected when the N reaches a bigger value. After the maximum, the CPU utilization should decrease over N. But the VM cannot handle more process-5 than 21. So the expected maximum is not shown in the above graph.

The efficiency of the computer will increase with the multiprogramming at the beginning. But since the power of the CPU and memory is limited, the efficiency decrease with too much multiprogramming. Not all processes can fit into the memory at the same time, so pages will be swapped out and in, which deeply harm the overall performance.