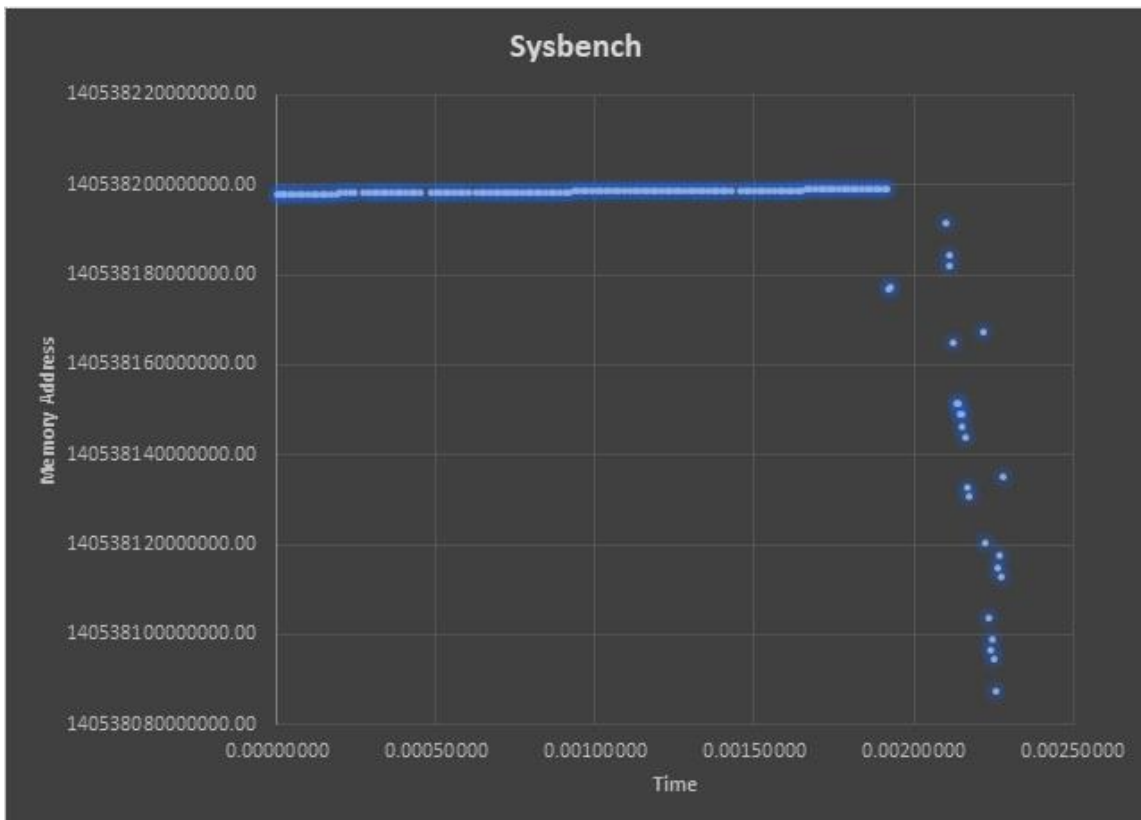
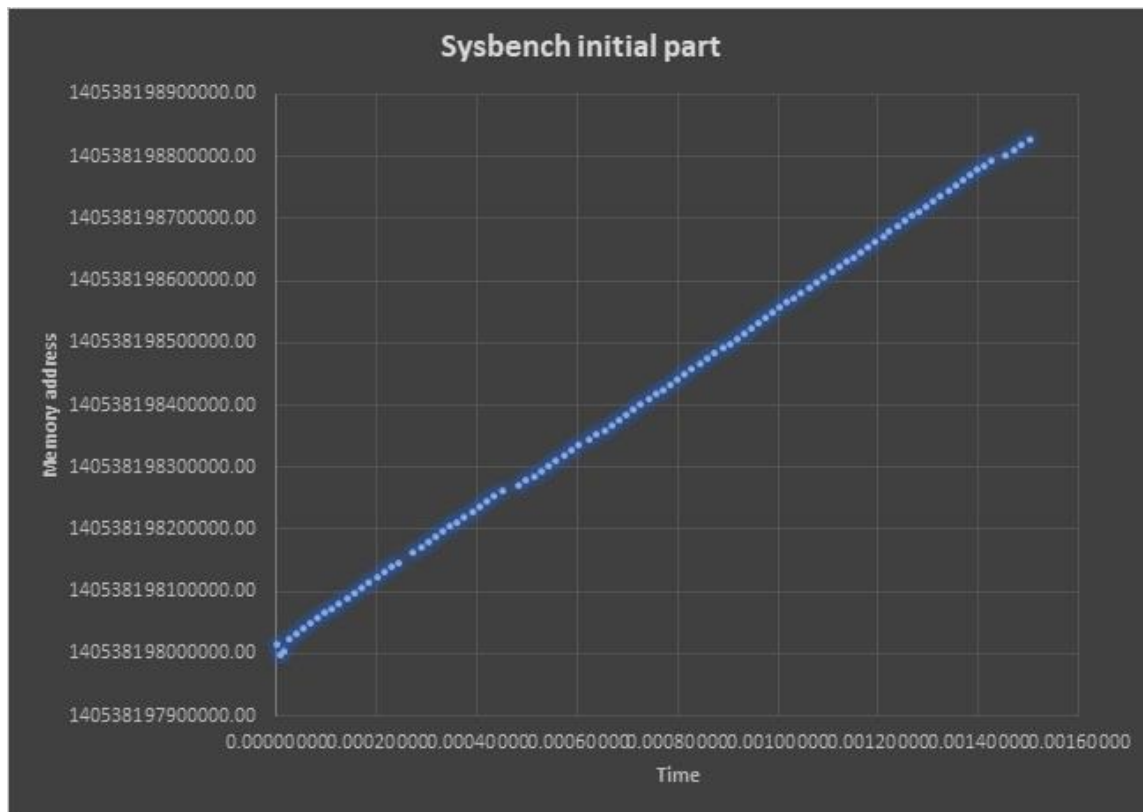


CS550 Operating Systems
Assignment-3 Report
Author- Rohit Mahendra Dhuri

Page faults for a compute intensive process



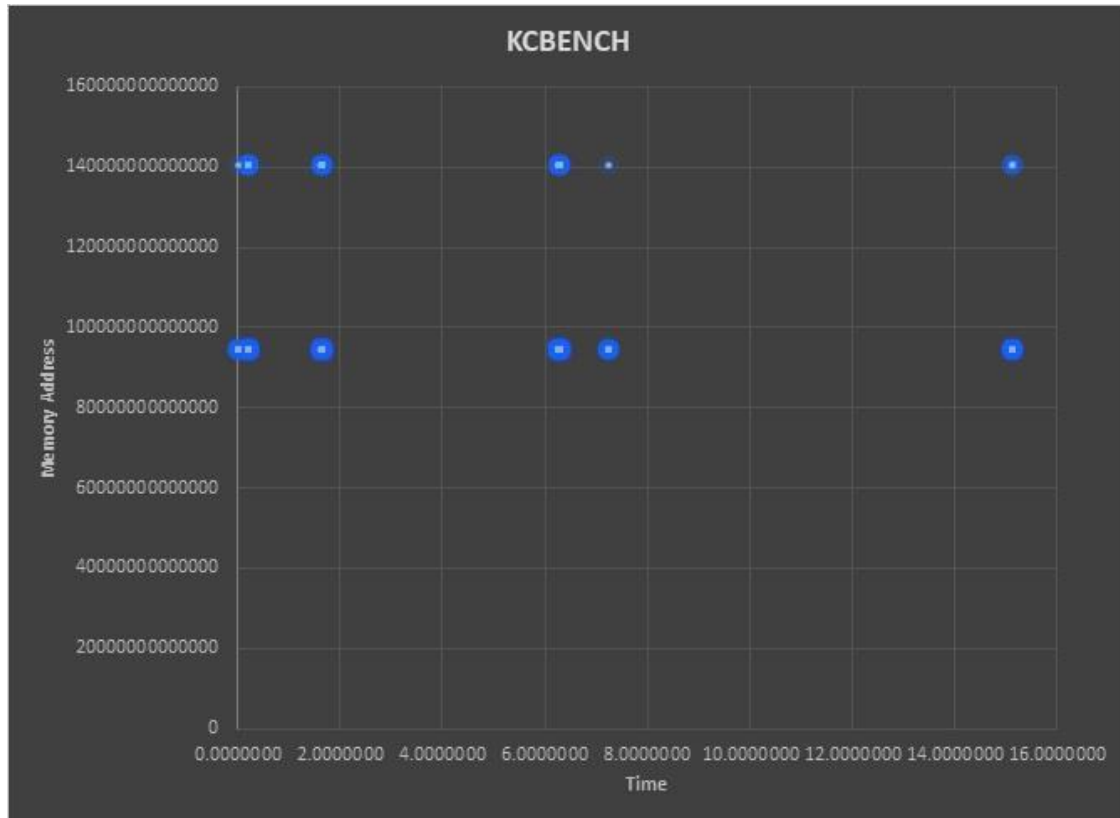
For a compute intensive process it can be observed that for most of the initial part of the process's life cycle the page faults start from a specific virtual memory address and then continue to occur on next increasing memory addresses. As in, in the scatter plot if you look closely a slope can be observed for the first 90% of values. Below is the plot for only the initial 90% values.



This shows that the initial part of the process's life cycle the page faults occur on increasing memory addresses as we go forward in time. And when the process is about to end page faults move to the initial part of the virtual address space for the process.

Benchmark details: Sysbench - <https://imysql.com/wp-content/uploads/2014/10/sysbench-manual.pdf>

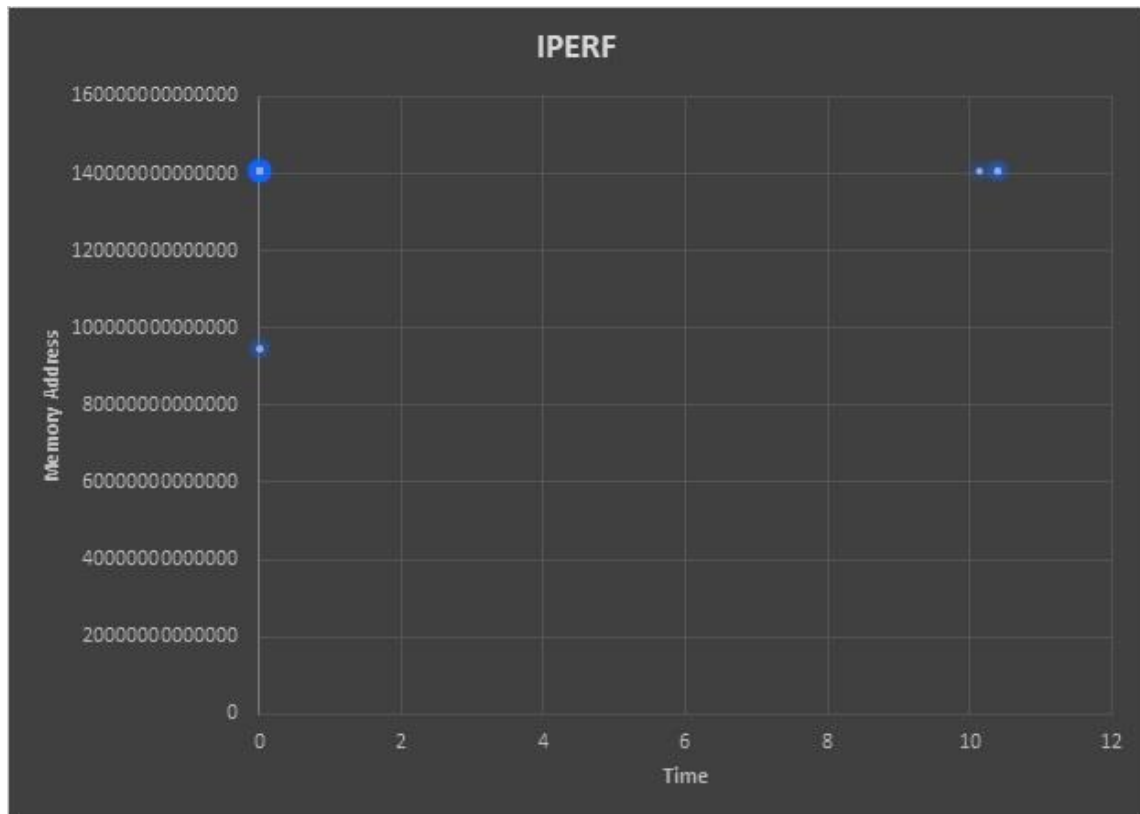
Page faults for a file IO Intensive process.



It can be observed that page faults continue to occur only around certain virtual memory addresses. Clusters of dots can be seen. This seems to be the case because a file may be distributed among contiguous memory addresses. Hence the page faults only occur in the virtual memory address space of these files. And the system switches between doing IO operations on different files.

Benchmark details: kcbench - <https://gitlab.com/knurd42/kcbench>

Page faults for a network Intensive process.



For using Iperf benchmark we used two systems- a client system and a server system. The krpobe was installed on the client system and it tracked the client process. Clusters can be seen in the scatter plot. Similar to the one in kcbench. But, not as many since this benchmark did not run for a longer time it may not have operated on a memory address space of the size of kcbench. Here, it pings the server and sends and receives memory packets. It must be sending packets corresponding to a single file and also receiving packets correspond to a single file, hence forming the clusters in the graph.

Benchmark details: Iperf - <https://iperf.fr/>