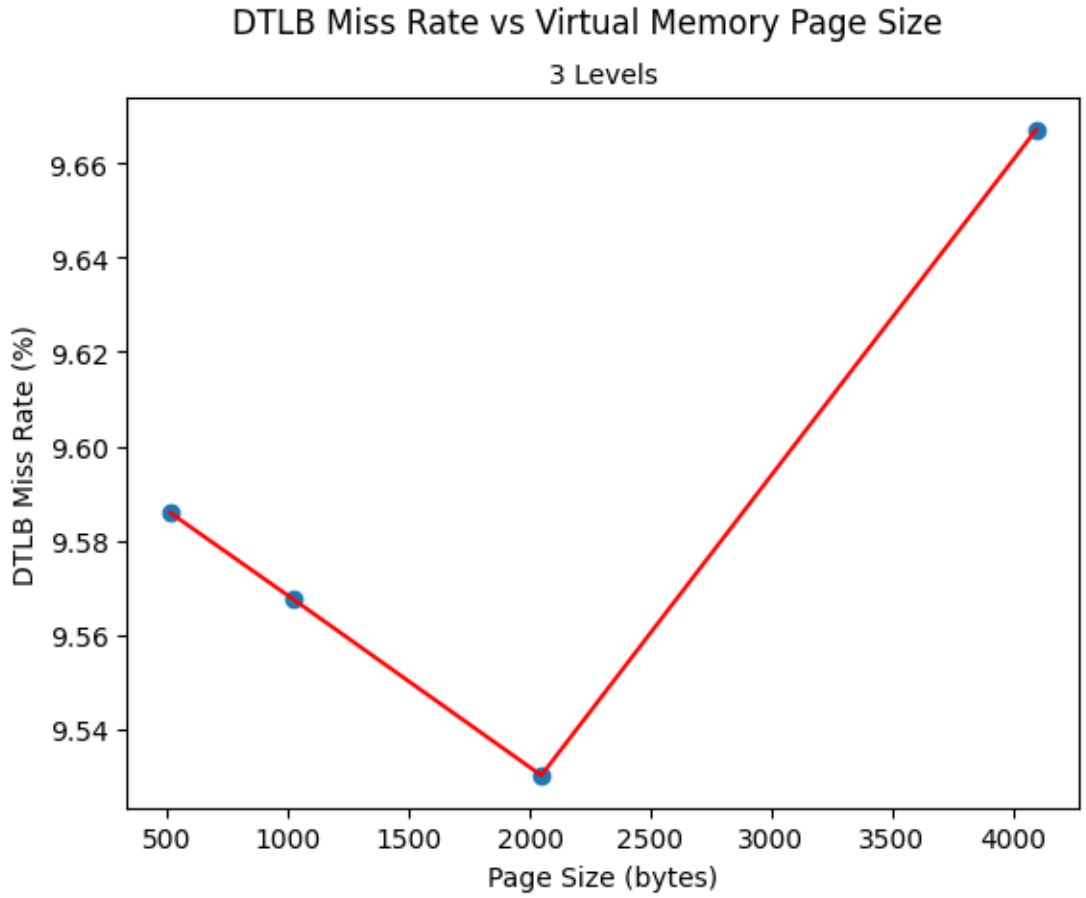
P10) Plot the DTLB Miss Rate by varying the Virtual Memory size keeping the Physical Page size constant (at least 4 different sizes of page table). Also vary the number of levels of tables (at least 2 values including default). Analyze the result.

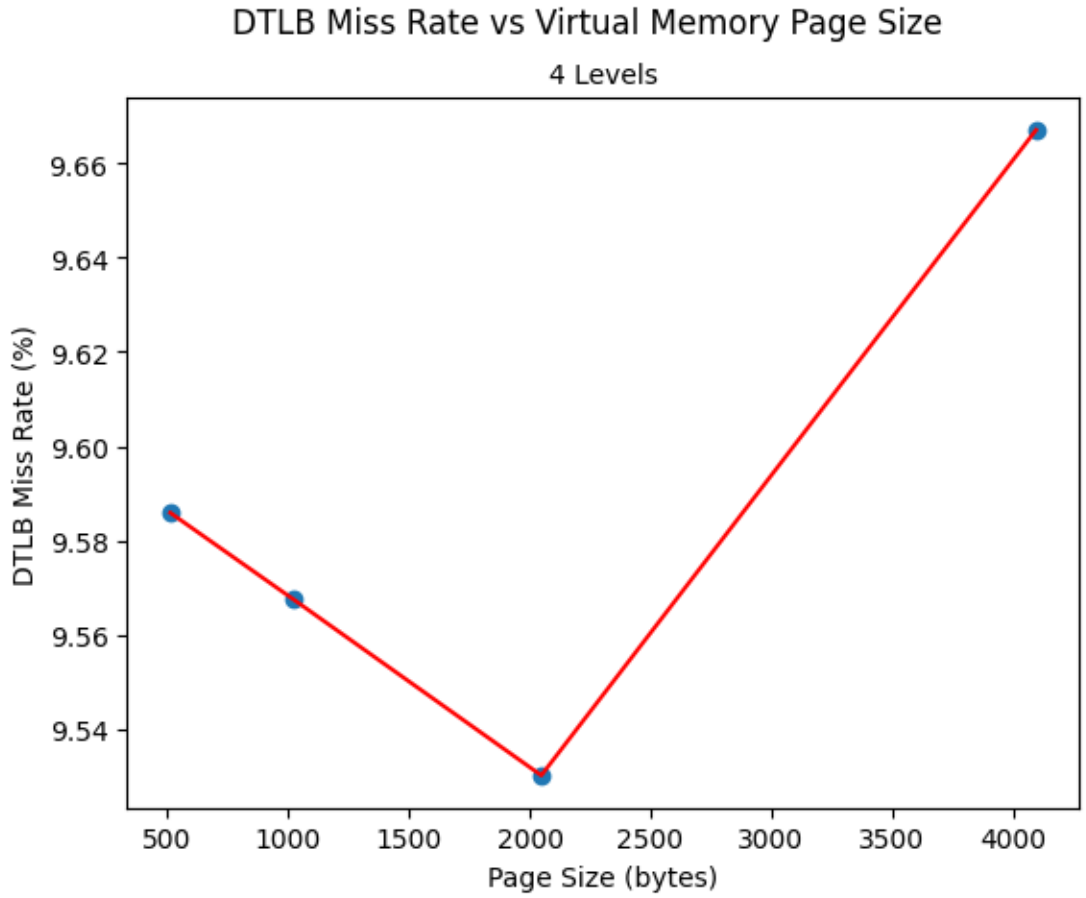
The following graphs were plotted for 4 different page table sizes: 512, 1024, 2048 and 4096 bytes. The graphs are a plot of the DTLB miss rate versus the Page Size.

3 Levels of Tables



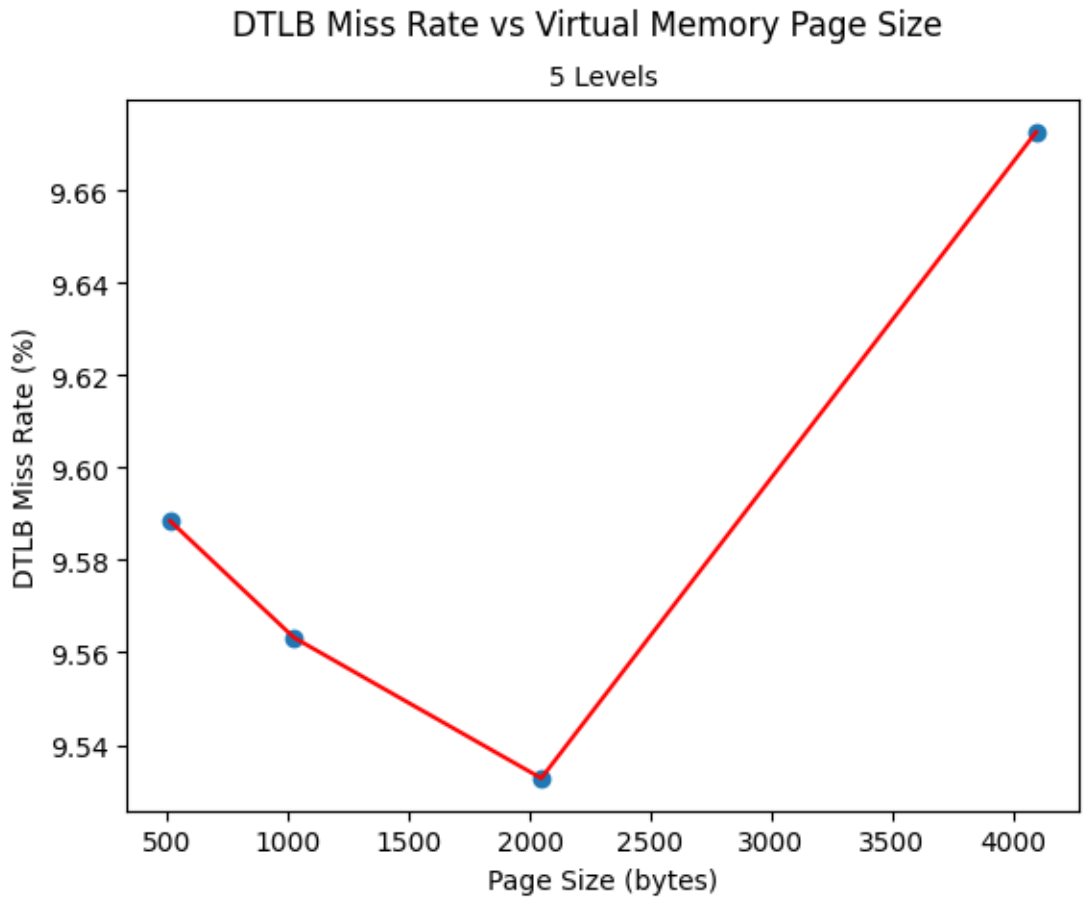
Page Size (bytes)	Number of Accesses	Number of Misses	Miss Rate (%)
512	21305962	2042365	9.5859
1024	21346879	2042348	9.5674
2048	21358715	2035519	9.5302
4096	21304852	2059561	9.6671

4 Levels of Tables



Page Size (bytes)	Number of Accesses	Number of Misses	Miss Rate (%)
512	21305962	2042365	9.5859
1024	21346879	2042348	9.5674
2048	21358715	2035519	9.5302
4096	21304852	2059561	9.6671

5 Levels of Tables [Default]



Page Size (bytes)	Number of Accesses	Number of Misses	Miss Rate (%)
512	21303662	2042689	9.5884
1024	21350854	2041826	9.5632
2048	21358755	2036076	9.5327
4096	21308911	2061138	9.6727

Analysis

Effect of Page Size on DTLB Miss Rate

Normally, we would expect the DTLB miss rate to decrease with increase in page size. This is because larger the page size, more the data available at once and lesser chance for a page fault. When the page size is increased, more memory is brought into the TLB with each page fault. This means that the TLB will contain a larger chunk of the program's address space. As a result, if the program exhibits spatial locality, it's more likely that the recently accessed pages and their neighboring pages are already in the TLB. This reduces the chance of a page fault.

While the above argument demands that we have a monotonically decreasing curve for the TLB miss rate versus page size curve, we observe an anomaly in the above graphs. While the TLB miss rate decreases from page size 512 bytes to page size 2048 bytes, there is a sudden increase when the page size is increased to 4096 bytes.

1. The simplest argument for this would be that since the page size is increasing, beyond a certain size, a page replacement due to a page fault could potentially remove an address that would be accessed in the near future, causing another page fault, and in turn result in a larger TLB miss rate.

2. Another argument for the same could be Belady's Anomaly. If the page replacement policy is not a stack page replacement policy, i.e. FIFO, Random Page Replacent Algorithm, etc, this Anomaly occurs, causing a sudden increase in the number of page faults and in turn, the miss rate.

Effect of Number of Levels of Tables on DTLB Miss Rate

A single-level page table could potentially store the entire mapping of virtual to physical memory addresses in one table which reduces the chance of a TLB miss. For a multi-level page table, the TLB entries may need to be updated for multiple levels of page table lookups, which could increase the miss rate.

Cache Configuration

```
infernape@MSI:~/ChampSim$ getconf -a | grep CACHE
LEVEL1_ICACHE_SIZE                 32768
LEVEL1_ICACHE_ASSOC
LEVEL1_ICACHE_LINESIZE             64
LEVEL1_DCACHE_SIZE                 32768
LEVEL1_DCACHE_ASSOC                8
LEVEL1_DCACHE_LINESIZE             64
LEVEL2_CACHE_SIZE                  262144
LEVEL2_CACHE_ASSOC                 4
LEVEL2_CACHE_LINESIZE              64
LEVEL3_CACHE_SIZE                  12582912
LEVEL3_CACHE_ASSOC                 16
LEVEL3_CACHE_LINESIZE              64
LEVEL4_CACHE_SIZE                  0
LEVEL4_CACHE_ASSOC
LEVEL4_CACHE_LINESIZE
```