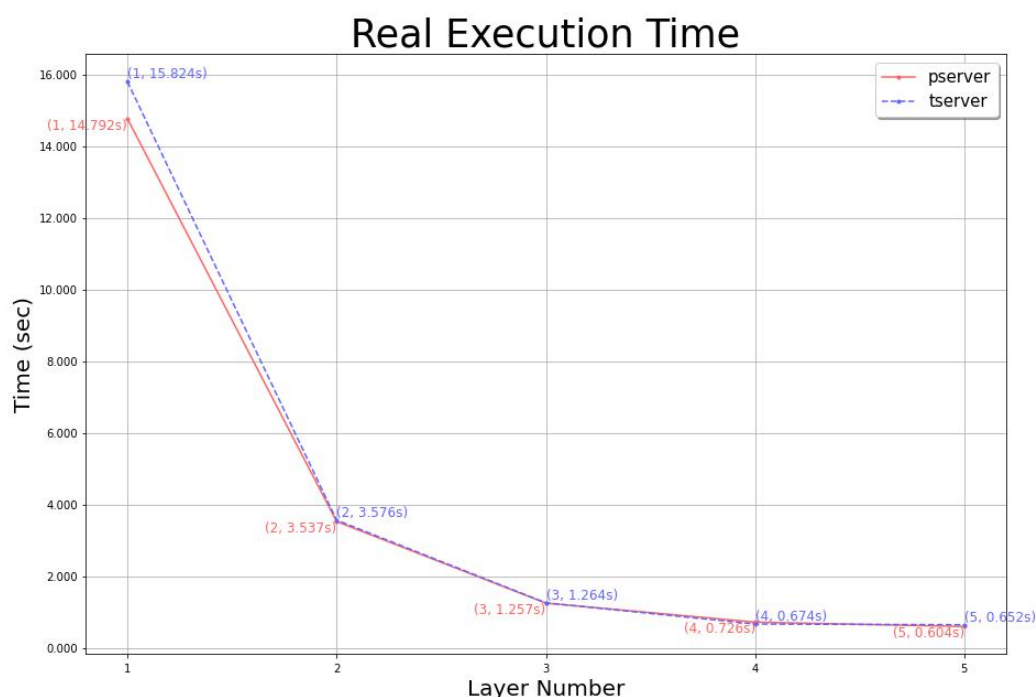


SP Prog. 4 Report

B08501098 柯晨緯

Server \ Layer	1	2	3	4	5
pserver	14.792s	3.537s	1.257s	0.726s	0.604s
tserver	15.824s	3.576s	1.264s	0.674s	0.652s



上方圖表是在 linux2.csie.ntu.edu.tw 上測試的。每個Layer Number對應的數據皆是重複測試5次後，取平均得到的結果。

首先我們能看到當降低Layer的數量後，會花較多時間去執行。這是因為Layer數量下降代表mergesort的basecase中要sort的數量上升，又因為助教實作的 `sort()` 為一複雜度為 $O(N^2)$ 的算法，因此會造成執行時間急遽上升。接著，當Layer數量增加後，我們可以看見其執行時間下降的幅度趨緩。這是因為Layer數量增加後，系統會需要製造較多的thread/process來工作，多出額外的overhead，造成優化的幅度越來越少。

我的實驗中，在Layer數量為1時，才可以較明顯看出 `tserver` 需要花的時間更久，否則在Layer數量大於1後，兩者的運行時間相差不多。

值得注意的是，我的 `tserver` 中存取movie titles的實作方式為移動指標而已，而 `pserver` 則是用 `strncpy()` 來複製電影名稱。因此我認為在單一request下，用multiprocess加上shared memory來處理會比multithread來的快。