

Computer Organization Lab6

0816146 韋詠祥

Fixed Associativity

在固定 Associativity 為 1 (direct map cache) 時，從實驗數據來看，這筆測試資料中 cache size 增加到 32 KB 就達到非常好的表現，就算跟 512 KB 比起來差距也在可以接受的範圍內。

就 block size 來說，當 cache size 愈大時，block size 也需要跟著變大才會出現最佳效果，大致上 64 bytes 可以取得不錯的表現。

在 cache size 為 4 KB 時，最佳的 block size 為 16 bytes，與其他 cache size 時的趨勢不太相同，推斷是因為當 cache 不夠大時，增加 block size 會減少可以放的 block 數量，在 Spatial locality 不夠明顯的前提下，反而排擠到某些原本不該 miss 的 cache。

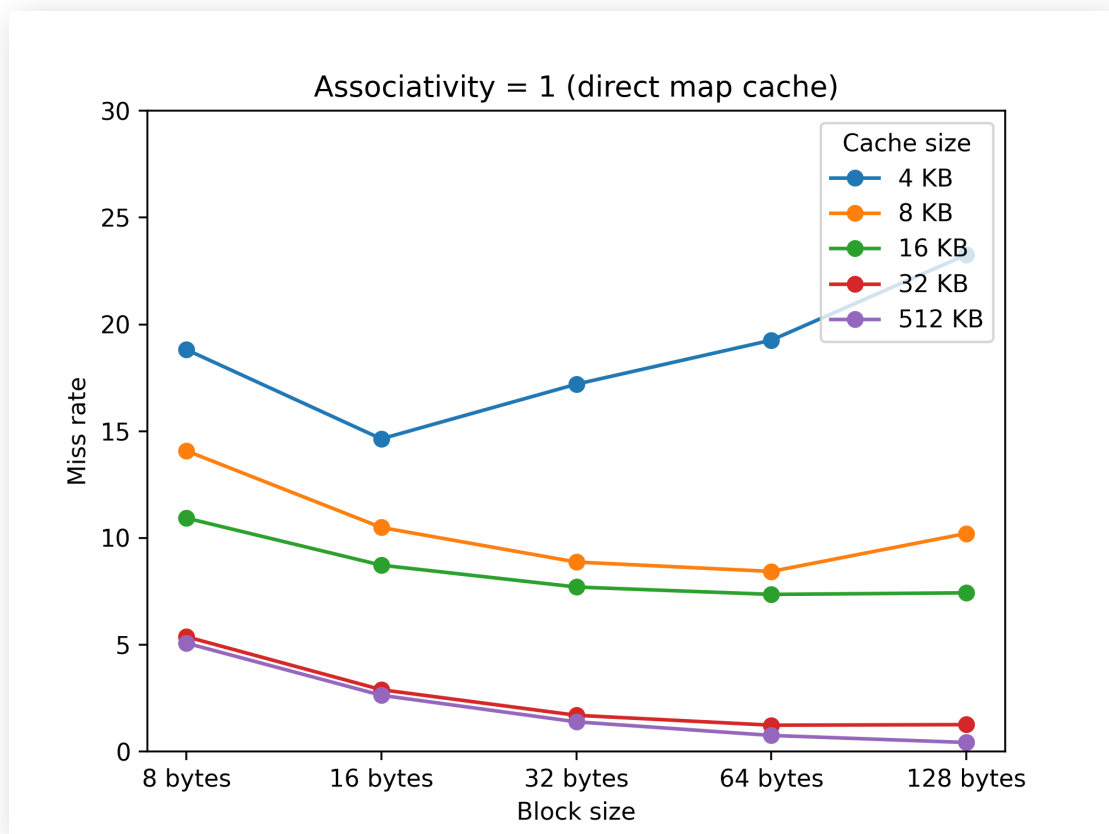


Fig 1. 固定使用 direct map cache 繪製關係圖

Fixed Block Size

當固定 block size 為 32 bytes 時，可以看到 associativity 愈高，實驗的結果會愈好，不過這是只考量到 cache miss rate 的理論值，尚未考量到 n-way 甚至是 fully associative 造成的成本增加、效能損耗。

從這次實驗的結果來看，使用 2-way set associative 帶來的效能改進已經很明顯的，到 4-way 接近完美，與 8-way 或 32-way 相距不遠。在尚未詳細考量實作、硬體成本的前提下，我認為視情況選擇 2-way 或是 4-way 最為恰當。

對於不同 cache size 的比較，很明顯的 miss rate 從 1 KB 到 32 KB 呈現單調降低，這是因為 cache size 愈大，在發生 cache miss 時愈不會將未來還需要用到舊資料覆寫。

而 cache size 增加到 32 KB 後，已經與 512 KB 非常接近，推測可能是因為這筆測試資料所包含的位址範圍就這麼大，當 cache 達到足夠大時，cache miss 只發生在第一次抓取資料，後續不會因為 cache 不夠用而覆寫過期的舊資料。

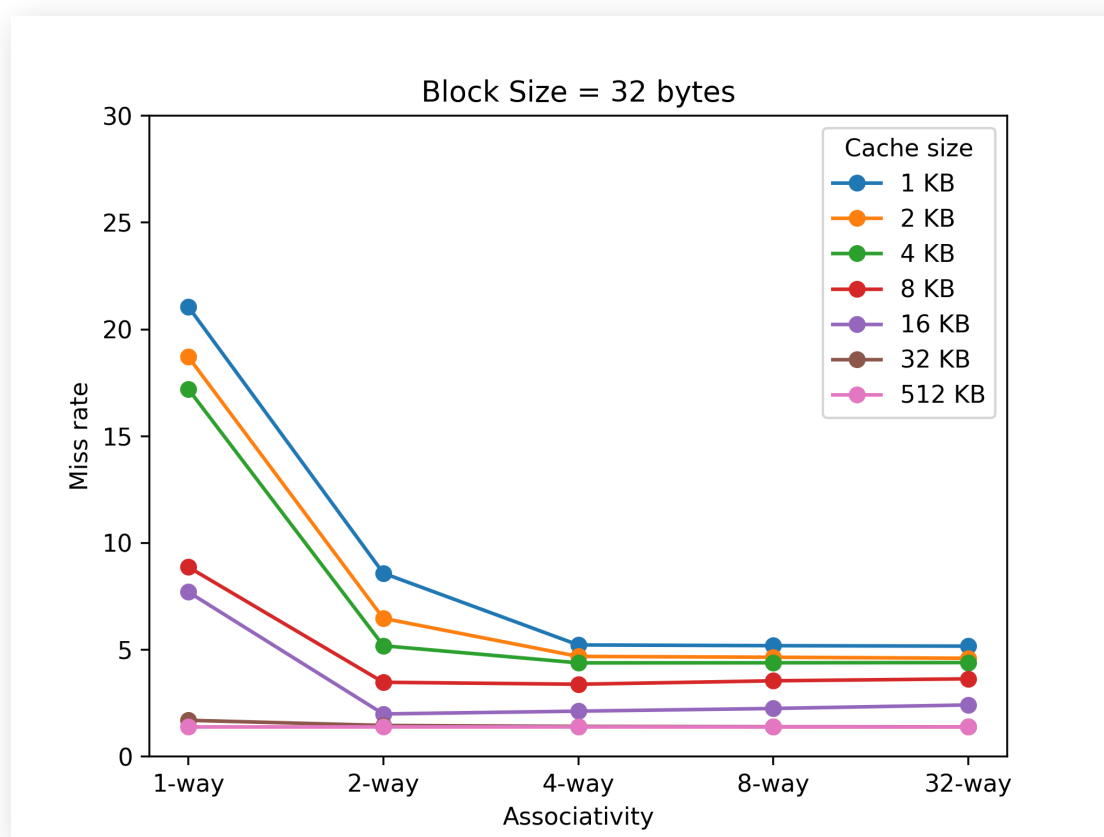


Fig 2. 固定使用 block size = 32 bytes 繪製關係圖

Problems You Met and Solutions

在開始做之前，直覺以為 n-way set associative 會很難實現，不過把 direct map cache 的程式碼理解並修改完成後，才發現這不過就加幾行的事情而已。在 LRU 的實作用到了相對高階的判斷方式，在硬體實作相對不可行，真的只能拿來作為模擬用途。

完成程式碼後，在決定使用哪套工具繪製圖表也花了點時間研究，最後選擇使用 Python 搭配 MatLab Plot Library，兩張圖片各用了 20 行程式碼，最終成果的 plot.py 也才 60 行，比想像中簡單很多。

Summary

經過這次 Lab 6 的練習，讓我更瞭解 n-way cache 與 LRU 的運作機制，也透過 associativity、block size、cache size 三個參數的操作，大致上將這三個參數如何互相影響弄清楚了。

最後有試著想做 fixed cache size 的實驗，不過在不同 cache size 下，得出的結果也會有不小的差異，較難得出一個合理的結論，後來因而作罷。

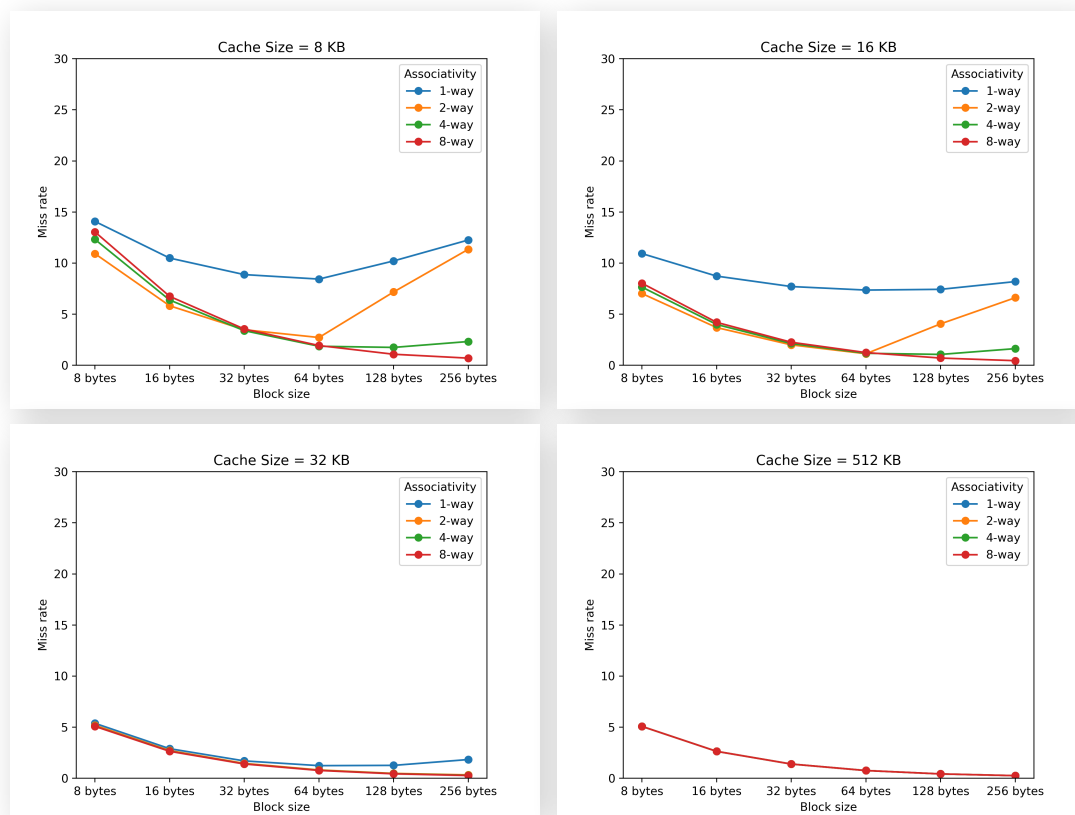


Fig 3. 固定 cache size 繪製關係圖