

平行程式 Parallel Programming – HW4: Blocked All-Pairs Shortest Path

姓名：蕭子馨 學號：103062372

1、Implementation

- (a) IO 的部分使用 c++ 的 `ifstream` 與 `stringstream` 的 `rdbuf` 讀取整個檔案，之後利用 `char` 指標掃過一遍整個 `string` 來分析並轉換成 `int`，藉此得到最優化的速度。
- (b) Cuda 的部分，我是使用標準的 32×32 的 `block` 來做計算，沒有做特殊的形狀分割或是任何 `stream` 優化。總共分成 3 個 `phase`，第一個 `phase` 計算 `Pivot block`，第二個 `phase` 則是將 `row` 與 `column` 合併在 `kernel` 裡面區分。第三的 `phase` 則將所有的 `block` 都包成一份 `kernel` 來並行執行，並在 `kernel` 內判斷是否是 `phase 1` 或 `2` 時計算過的 `block`，是則結束 `block`。
- (c) Openmp 的部分，我是將一整個完整的 `adjacent matrix` 切成 2×2 的 `block`，再將 `block` 分別平行到兩張 GPU 上，詳細過程如下：將 `Block` 分別定義為 `G11`、`G12`、`G21`、`G22`。則兩個 `gpu` 執行的流程分別如下：

Stream 0	Stream1
G11	G11
G12	G21
-----sync	
G22	G22
G12	G21
-----sync	
G11	G11
-----sync to host	

利用這個方法，能夠將原本需要每一回合同步的次數縮減到只需要 3 次，雖然仍有大約 $2/3$ 的 `block` 重疊到，但是已經足以得到比單片 GPU 佳的效果。因此就沒有再繼續優化了。

(d) 努力加載中...

2、Profiling Results

3、Experiment & Analysis

4、Conclusion