Matrix Multiplication with Multiple Devices

2014-12472

# 1. Objective

OpenCL을 사용하여 여러 계산 장치들을 이용하여 다양한 기기의 커널 작성과 작업 분배를 익힌다.

# 2. Approach

하나의 커널을 이용하여 거의 동일한 양의 계산을 각 장치에 분배한다.

## 2.1 2GPUs

행렬을 가로로 반을 잘라 위 아래 분배하였다. 가로로 자른 이유는 spatial locality가 더 좋기 때문이다.

## 2.2 4GPUs

행렬을 가로로 4등분 하여 각각 분배하였다.

## 2.3 1CPU, 1GPU

7:1 비율로 GPU가 더 많은 일을 하도록 분배하였다. 주어진 커널이 CPU같은 장치에선 성능이 좋지 않아서 그렇게 하였다.

## 2.4 1CPU, 4GPU

마찬가지로 7:1 비율을 택했다.

# 3. Compile & Execution

make를 하면 6개의 binary를 만들어 낸다. main함수에 바뀐 것이 없기에 option과 행렬 크기는 전 과제들과 똑같이 주면 된다.

# 4. Results

|  |  |
| --- | --- |
| 장치 | 시간 |
| GPU | 8.366689 sec |
| GPU2 | 5.131406 sec |
| GPU4 | 3.861648 sec |
| CPU | 979.635807 sec |
| CPUGPU | 121.831152 sec |
| CPUGPU4 | 121.411644 sec |

CPU커널이 느려서 CPU커널 실행 시간이 대부분의 실행 시간을 차지한다. GPU들만 이용했을 경우 정확히 반 만큼은 아니지만 거의 반 씩 줄어드는 것을 볼 수 있다.

CPU의 실행 시간을 줄이는 방법으로는 B행렬을 transpose 하는 등 spatial locality를 좋게 하는 방법이 있을 것으로 예상한다. 또 더 작은 크기의 work group size가 한 work group 실행의 context switch 비용을 줄일 수 있을 것으로 기대된다.