

과제 #2

4190.414A 멀티코어 컴퓨팅(001)

Due: 2020년 4월 21일(수) 23시 59분

1 문제 1: pthread를 이용한 병렬화

행렬 곱셈($A_{M \times K} \times B_{K \times N} = C_{M \times N}$)을 수행하는 예시 코드가 주어진다. 예시 코드는 `make` 명령으로 컴파일이 가능하며, `./main -h` 명령으로 지원하는 옵션을 확인할 수 있다. 다음은 실행 예시이다.

```
$ ./main -v -n 3 4096 4096 4096
Options:
  Problem size: M = 4096, N = 4096, K = 4096
  Number of threads: 1
  Number of iterations: 3
  Print matrix: off
  Validation: on

Initializing... done!
Calculating...(iter=0) 13.103662 sec
Calculating...(iter=1) 13.088932 sec
Calculating...(iter=2) 13.042679 sec
Validating...
Result: VALID
Avg. time: 13.078424 sec
Avg. throughput: 10.508831 GFLOPS
```

`-t` 옵션으로 지정한 스레드의 개수가 `mat_mul` 함수의 `_num_threads` 인자로 주어진다. 예시 코드는 주어진 값에 상관없이 1개의 스레드만을 생성하여 행렬 곱셈을 수행하도록 구현되어 있다. 조교가 기초적인 병렬화를 적용한 프로그램 `main_ref`은 여러개의 스레드를 사용하면 성능이 향상되는 것을 볼 수 있다. `main_ref`의 구현은 비밀이며 성능 참고용으로 실행 파일만 주어진다.

```
$ ./main -t 40 -n 3 4096 4096 4096
Avg. time: 13.138292 sec
Avg. throughput: 10.460945 GFLOPS # 성능에 차이 없음
$ ./main_ref -t 40 -n 3 4096 4096 4096
Avg. time: 1.180623 sec
Avg. throughput: 116.412254 GFLOPS # 빨라짐
```

예시 코드를 수정하여 여러 개의 스레드를 사용하도록 병렬화하라. 주의 사항은 다음과 같다.

- `mat_mul.c`와 `Makefile`만 수정할 수 있다. 다른 파일은 채점 시에 예시 코드로 덮어씌워진다.
- `pthread` 이외의 병렬화 방식(`OpenMP`, `MPI` 등)은 사용을 금한다.

- Vector instruction은 사용 가능하다.
- 반드시 주어진 `_num_threads` 이하의 스레드를 사용하도록 구현하여야 한다. 몰래 더 많은 스레드를 사용해서 성능의 이득을 본 경우 0점 처리된다.
- 본 과제는 반드시 실습 서버의 계산 노드 (c3, c4, c5) 에서 진행하도록 한다.
- 사용한 스레드 개수를 셀 때, 계산을 수행한 스레드만 개수를 센다. 예를 들어, 예제 코드는 메인 스레드 1개 + 생성된 스레드 1개, 총 2개의 스레드를 사용하지만 메인 스레드에서는 계산을 하지 않으므로 1개의 스레드로 취급한다. 만약 제출한 코드가 메인 스레드에서 계산을 수행한다면 메인 스레드도 개수에 포함된다.

보고서에는 다음 내용이 들어가면 좋다.

- 병렬화 방법에 대한 설명
- 스레드 개수에 따른 성능 변화 및 분석
- Theoretical peak FLOPS와의 비교
- 실습 서버의 CPU는 Lecture 7에서 배운 SMT가 적용되어 physical core는 40개이지만 logical core는 80개이다. 스레드를 80개 사용하였을 때, 1 - 40개의 스레드를 사용하였을 때와 비슷한 추세로 성능이 증가하는가?

채점 기준은 다음과 같다.

보고서 (40%)

정확성 (40%) 40개 이하의 스레드, 4096 이하의 M , N , K 에 대해서 `-v` 옵션을 통한 validation을 통과해야 한다.

성능 (20%) 실습 서버에서 40 스레드, $M = N = K = 4096$ 옵션을 주고 실행했을 때, 150 GFLOPS를 넘으면 만점. 그 이하는 비율에 따라 점수를 부여한다. (e.g., 135 GFLOPS인 경우 성능 점수의 90%를 부여) 답이 틀린 경우 0점.

2 문제 2: 캐시의 크기 추정

시스템의 캐시 크기를 모른다고 가정하자. 문제 1에 주어진 행렬곱 프로그램으로 캐시의 크기를 추정하는 방법을 고안하고, 그 방법으로 시스템의 캐시 크기를 추정해본 후 결과를 논의하라. 행렬곱 프로그램은 주어진 예시 코드를 사용해도 좋고 본인이 병렬화한 프로그램을 사용해도 좋다.

3 제출 방법

- 조교 메일(jinpyo@aces.snu.ac.kr)로 보고서를 포함한 모든 파일을 하나의 파일(e.g., .zip, .tar.gz)로 압축 후 첨부하여 제출한다.
- 메일 제목은 계정이름.HW2으로 한다. (e.g., mc99.HW2)
- 첨부파일명은 계정이름.HW2.확장자으로 한다. (e.g., mc99_HW2.zip, mc99_HW2.tar.gz)
- 제출한 메일은 기계적으로 처리되므로 위의 내용을 지키지 않을 시 누락될 수 있으니 잘 지켜주시기 바랍니다.
- Grace day를 사용하고자 하는 경우에는 메일 내용에 이를 반드시 포함한다.