

[CSEG437/CSE5437] 수치 컴퓨팅 및 GPU 프로그래밍

숙제 4

담당교수: 서강대학교 컴퓨터공학과 임 인 성

2018년 6월 11일

마감: 6월 26일(화요일) 오후 8시 정각

제출물 및 제출 방법: 보고서 형식의 분석 내용과 작성한 프로그램 및 수행 결과 출력물 등.

[목적] 이번 숙제에서는 선형 방정식의 풀이 기법인 Jacobi 방법과 Gauss-Seidel 방법에 대한 이해를 높임을 목적으로 한다.

- 본 숙제에서 제공하는 선형 방정식 데이터를 사용하여 주어진 sparse, symmetric, and row diagonally dominant linear system $Ax = b$ 의 해를 구하여 보자. 이 데이터들은 각각 컴퓨터 그래픽스 분야에서 자연스러운 물 애니메이션 효과를 생성하기 위하여 널리 쓰이고 있는 Navier-Stokes 방정식의 풀이 과정에서 발생한 선형 방정식으로서, 다음 논문의 내용을 구현할 때 생성한 데이터이다.

<ftp://grmanet.sogang.ac.kr/pub/ihm/webpapers/sca06.pdf>

실제로 본 유체 시뮬레이션 과정에서는 PCG (Preconditioned Conjugate Gradient)와 같은 고급 방법을 사용하였으나, 본 숙제에서는 Gauss-Seidel 방법과 Jacobi 방법을 고려한다.

- 본 과목에서 제공하는 데이터 파일 (mat****.txt)에 기술된 선형 방정식 $Ax = b$ 의 A 행렬은 모두 *sparse*하고 *symmetric*하고 *diagonally row dominant*함.
- 데이터 파일의 저장 포맷은 readme 파일에 기술되어 있는데, 특히 본 숙제에서 다루는 *large, sparse, and symmetric* 행렬을 어떠한 자료 구조를 사용하여 읽어 들일 지를 잘 생각한 후, 효율적인 프로그램이 되도록 노력할 것. 특히 단순히 배열을 잡을 때보다 얼마나 많이 메모리를 절약할 수 있는지 고려하라.

[문제 1] <http://www.netlib.org>에 공개되어 있는 코드 중, SLATEC 패키지의 Gauss-Seidel 기법 구현 코드인 `dsgs.f` 함수를 사용하여 선형 방정식의 근을 구하라. (조교는 필요시 이 파일을 컴파일한 obj 파일을 제공할 예정임)

- 숙제를 하기 위하여 데이터 파일에서 선형 방정식 $Ax = b$ 의 A 와 b 를 읽어 들여 적절한 초기 벡터 $x^{(0)}$ 에 대해 Gauss-Seidel 방법을 통하여 근에 대한 근사 값 $x^{(n)}$ 을 구한 후, 데이터 파일에

마지막에 저장되어 있는 근 x 과 얼마나 유사한지 (즉 $e = x - x^{(n)}$ 와 $r = Ax - Ax^{(n)}$ 에 대한 상대오차의 크기가 얼마인지) 확인하고 그 결과를 분석하여 제출하라. 특히 다양한 반복 회수 n 에 대해 위의 오차를 분석해봄으로서, Gauss-Seidel 기법이 위의 문제에 대하여 얼마나 빨리 수렴을 하는지 분석하라.

[문제 2] 다음 CPU 상에서 Jacobi 방법을 C/C++ 언어를 사용하여 구현한 후, 본 숙제의 선형 방정식의 근을 올바르게 구하는지 확인하라.

- 이 경우 sparse symmetric matrix를 어떠한 형태의 자료구조를 사용하여 CPU 메모리에 저장할 지 Jacobi 방법의 특성을 고려하여 잘 설계할 것.
- (경우 1) 자신이 작성한 Jacobi 방법이 올바르게 근을 구할 경우
 - Gauss-Seidel 방법 대비 Jacobi 방법의 수렴 속도를 비교 분석하라.
 - 이 경우 두 방법 모두 동일한 초기값을 사용해야 하며, 어떠한 초기값에 대하여 실험을 할 지 잘 고려할 것.
- (경우 2) 자신이 작성한 Jacobi 방법이 올바르게 근을 구하지 못할 경우 (올바르게 근에 수렴할 것으로 예상되나)
 - 해당 행렬 A 의 대각선 원소 a_{ii} 값에 1을 증가시킨 후, 원래의 근 x 를 사용하여 $b = Ax$ 를 구해 다시 자신의 Jacobi 방법을 사용하여 선형 방정식 $Ax = b$ 의 근 x 를 구하라.

[문제 3] (위의 두 경우 중 해당하는 경우에 대해) 다음 GPU 상에서 Jacobi 방법을 OpenCL을 사용하여 구현한 후, 본 숙제의 선형 방정식의 근을 올바르게 구하는지 확인하라.

- 이 경우 sparse symmetric matrix를 어떠한 형태의 자료구조를 사용하여 GPU 메모리에 buffer object 형태로 저장할 지 Jacobi 방법의 특성을 고려하여 잘 설계할 것.
- (비록 본 숙제의 선형 방정식의 크기가 아주 크지는 않으나) CPU와 GPU 각각에서 자신이 구현한 Jacobi 방법의 속도를 비교하여 과연 GPU 상에서 가속이 잘 되었는지 확인하라. GPU 시간은 순수하게 GPU 상에서 시작하여 끝날 때까지의 시간만 측정하라.
- 의미있는 시간 비교를 위하여 필요하다면 반복적으로 근을 구해 실험을 해볼 것.

- [주의]
1. 본인이 본 숙제에서 요구하는 것 중 무엇을 어디까지 완성했는지 README 파일에 정확히 기술하라.
 2. 실험을 통하여 발견한 사실이나 기타 특이 사항을 구체적으로 기술하라.
 3. 제출 방식은 조교가 과목 게시판에 공고할 예정임.
 4. 제출 화일에서 바이러스 발견 시 **만점 X (-1)**이고, 다른 사람의 숙제를 복사할 경우 관련된 사람 모두에 대하여 **만점 X (-10)**임.