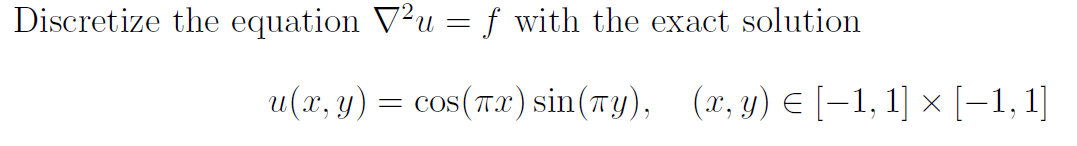
HW #2

2013-20968 나종걸 / 화학생물공학부



Solve this by jacobi, and parallel jacobi iter. , red-black gauss-seidel iter., with grids number Nx=Ny=100

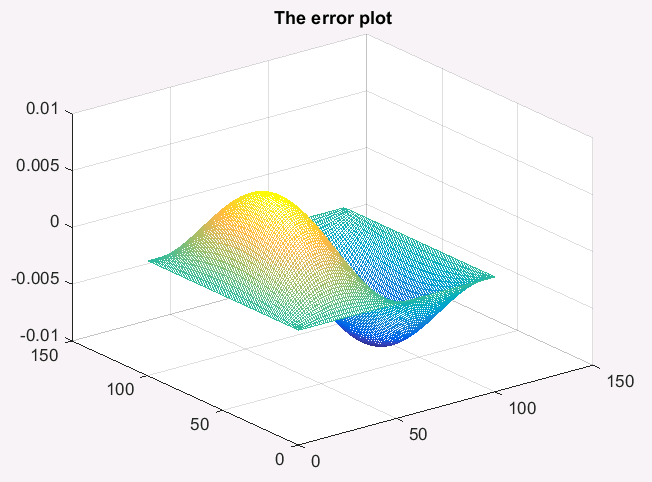
Source file은 server에 올려두었음.

그냥의 경우 iteration 을 100000 번 했을 경우

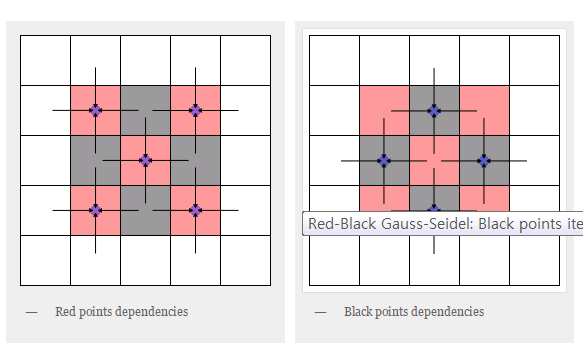
 cpu time이 7.89초

Parallel의 경우

로 3.79초로 약간 빨라진 것을 볼 수 있으나, 자체가 parallel을 지원하는 것 형태가 아니므로 red-black gauss-seidel로 해야 의미가 있을 것임 output을 matlab으로 그려보면 좋은 결과가 나오는 것을 볼 수 있다.



Red-black gauss seidel iteration의 경우 아래 아래와 같이 red dependence한 부분과 black dependence한 부분으로 나눠서 코딩을 잘 했다고 생각되는데, cpu time이 기하급수적으로 늘어난다. 어디선가 false sharing이 일어 난 것 같은데 잘 모르겠다.



for (int p=1; p<k; p++)

{

#pragma omp parallel for num\_threads(24)

for (int i=2; i<=n; i++)

{

for (int j=(i%2) + 2; j<=n; j+=2)

{

u2[i][j] = (u2[i-1][j]+u2[i+1][j] + u2[i][j-1] +u2[i][j+1] ) \*0.25 - h1\*f[i][j]/4;

}

}

#pragma omp parallel for num\_threads(24)

for (int i=2; i<=n; i++)

{

for (int j=((i+1)%2) + 2; j<=n; j+=2)

{

u2[i][j] = (u2[i-1][j]+u2[i+1][j] + u2[i][j-1] +u2[i][j+1] ) \*0.25 - h1\*f[i][j]/4;

}

}

}