

GPU Homework1

——孙正忠

1、 Principle

In the paper of A fast parallel Gauss Jordan algorithm for matrix inversion using CUDA by Girish Sharma, it introduces a method to solve this problem easily.

$$[C'] = \left[\begin{array}{cccc|cccc} 1 & a_{12}/a_{11} & a_{13}/a_{11} & \cdots & \cdots & 1/a_{11} & 0 & 0 & \cdots & 0 \\ 0 & a_{22} - a_{21} \times a_{12}/a_{11} & a_{23} - a_{21} \times a_{13}/a_{11} & \cdots & \cdots & -a_{21}/a_{11} & 1 & 0 & \cdots & 0 \\ 0 & a_{32} - a_{31} \times a_{12}/a_{11} & a_{33} - a_{31} \times a_{13}/a_{11} & \cdots & \cdots & -a_{31}/a_{11} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2} - a_{n1} \times a_{12}/a_{11} & a_{n3} - a_{n1} \times a_{13}/a_{11} & \cdots & \ddots & -a_{n1}/a_{11} & 0 & 0 & \cdots & 1 \end{array} \right]$$

In the picture, we can see the process of Gauss Jordan algorithm.

2、 Functions:

In this homework, my program is consist of 7 functions. 4 of the 7 functions are about gpu.

Host function:

```
void matrix_read()
```

Effect: read in the matrix which is generated from matlab.

```
void savetofile(double *A, string s, int n, int h)
```

Effect: save the matrix we need to a txt file.

```
int main()
```

Effect: the entrance of our program.

Device function:

```
_global__ void nodiag_normalize(double *A, double *I, int n, int i)
```

Effect: normalize the elements which are not in the diagonal.

```
_global__ void diag_normalize(double *A, double *I, int n, int i)
```

Effect: normalize the elements which are in the diagonal.

```
_global__ void gaussjordan(double *A, double *I, int n, int i)
```

Effect: the realization of parallel matrix inverse using Gaussian Jordan Elimination.

```
__global__ void set_zero(double *A, double *I, int n, int i)
```

Effect: set the first line in every loop to zero.

3、Performance comparison

SCALE	CPU RUN TIME(ms)	GPU RUN TIME(ms)(blocksize = 8)	GPU RUN TIME(ms)(blocksize = 32)
20*20	0	1.05536	1.21587
50*50	1	2.66234	3.49654
100*100	8	6.58941	7.94698
200*200	39	20.369	25.4879
500*500	622	190.305	211.504
1000*1000	4963	1080.69	1322.97

4、Reference

Sharma G, Agarwala A, Bhattacharya B. A fast parallel Gauss Jordan algorithm for matrix inversion using CUDA[J]. Computers & Structures, 2013, 128(9):31–37.